

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E
AUTOMAÇÃO INDUSTRIAL

**EVOLUÇÃO DE REDES IMUNOLÓGICAS PARA COORDENAÇÃO
AUTOMÁTICA DE COMPORTAMENTOS ELEMENTARES EM
NAVEGAÇÃO AUTÔNOMA DE ROBÔS**

Roberto Michelan

Orientador: Prof. Dr. Fernando José Von Zuben
DCA/FEEC/Unicamp

Co-orientador: Prof. Dr. Maurício Fernandes Figueiredo
DIN/UEM – PR

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica.

Área de Concentração: Engenharia de Computação

Campinas – São Paulo – Brasil

Julho de 2003

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

Michelan, Roberto

M582e Evolução de redes imunológicas para coordenação automática de comportamentos elementares em navegação autônoma de robôs / Roberto Michelan.-- Campinas, SP: [s.n.], 2003.

Orientadores: Fernando José Von Zuben; Maurício Fernandes Figueiredo.

Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Robótica. 2. Robôs – Sistemas de controle. 3. Robôs móveis. 4. Sistema imune. 5. Algoritmos genéticos. I. Von Zuben, Fernando José. II. Figueiredo, Maurício Fernandes. III. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. IV. Título.

Banca Examinadora

Orientador: Prof. Dr. Fernando José Von Zuben
(DCA/FEEC/Unicamp)

Co-orientador: Prof. Dr. Maurício Fernandes Figueiredo
(DIN/UEM, PR)

1º Membro Interno: Prof. Dr. Leandro Nunes de Castro Silva
(DCA/FEEC/Unicamp)

2º Membro Interno: Prof. Dr. Marconi Kolm Madrid
(DSCE/FEEC/Unicamp)

1º Membro Externo: Prof. Dr. André Carlos Ponce de Leon Ferreira de Carvalho
(ICMC/USP – São Carlos, SP)

Agradecimentos

Ao professor e amigo Fernando José Von Zuben, pela orientação, dedicação, paciência e visão de mundo.

Ao professor e amigo Maurício Fernandes Figueiredo, pela dedicação e encaminhamento, desde a graduação, nas trilhas da ciência e da pesquisa, e pelas orientações, inclusive nutricionais.

Ao professor e amigo Leandro Nunes de Castro Silva, pioneiro na área de sistemas imunológicos artificiais, a qual sustenta as principais contribuições deste trabalho.

Aos meus pais, que sempre me incentivaram e me apoiaram em todas as decisões, atenuando com palavras confortantes a dor da saudade e da inexperiência.

Aos meus segundos pais, Tia Lourdes, Tio Fausto e Ângelo, que me ampararam e fizeram com que eu me sentisse em casa, acompanhando comigo cada uma das dificuldades e vitórias.

À minha irmã Analucia, ao meu cunhado Marcelo e aos meus sobrinhos Bruno e Heloisa, que preencheram minha vida com muita alegria.

À Patrícia Vargas – “*Chatícia*” –, que foi minha companheira em muitos trabalhos e com quem dei muitas risadas.

Aos meus amigos da UEM e pessoas envolvidas, de alguma maneira, nesse período da minha vida.

Aos amigos da Unicamp, que sempre tinham uma palavra de incentivo, me fazendo rir das dificuldades.

A todos meus mestres, que me tornaram a pessoa que sou hoje.

À Faculdade de Engenharia Elétrica e de Computação (FEEC) e ao Departamento de Engenharia de Computação e Automação Industrial (DCA), pelas instalações e condições de trabalho que muito contribuíram para o desenvolvimento da pesquisa.

À CAPES, pela concessão da bolsa de mestrado.

Ao CNPq, pela participação no Projeto REBEL (470442/01-8) que tornou possível o experimento prático com o Robô Khepera II.

À oportunidade de viver, crescer e estar aqui para agradecer.

Resumo

MICHELAN, R., **Evolução de redes imunológicas para coordenação automática de comportamentos elementares em navegação autônoma de robôs**. Campinas: FEEC, Unicamp, Julho de 2003. Dissertação de Mestrado – Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 122 p.

A concepção de sistemas autônomos de navegação para robôs móveis, havendo múltiplos objetivos simultâneos a serem atendidos, como a coleta de lixo com manutenção da integridade, requer a adoção de técnicas refinadas de coordenação de módulos de comportamento elementar. Modelos de redes imunológicas artificiais podem então ser empregados na proposição de um controlador concebido com base em um processo de mapeamento dinâmico. Os anticorpos da rede são responsáveis pelos módulos de comportamento elementar, na forma de regras do tipo <condição>-<ação>, e as conexões são responsáveis pelos mecanismos de estímulo e supressão entre os anticorpos. A rede iniciará uma resposta imunológica sempre que lhe forem apresentados os antígenos. Estes antígenos representam a situação atual capturada pelos sensores do robô. A dinâmica da rede é baseada no nível de concentração dos anticorpos, definida com base na interação dos anticorpos e dos anticorpos com os antígenos. De acordo com o nível de concentração, um anticorpo é escolhido para definir a ação do robô. Um processo evolutivo é então responsável por definir um padrão de conexões para a rede imunológica, a partir de uma população de redes candidatas, capaz de maximizar o atendimento dos objetivos durante a navegação. Resulta então um sistema híbrido que tem a rede imunológica como responsável por introduzir um processo dinâmico de tomada de decisão e tem agora a computação evolutiva como responsável por definir a estrutura da rede. Para que fosse possível avaliar os controladores (redes imunológicas) a cada geração do processo evolutivo, um ambiente virtual foi desenvolvido para simulação computacional, com base nas características do problema de navegação. As redes imunológicas obtidas através do processo evolutivo foram analisadas e testadas em novas situações, apresentando capacidade de coordenação em tarefas simples e complexas. Os experimentos preliminares com um robô real do tipo Khepera II indicaram a eficácia da ferramenta de navegação.

Palavras-Chave: navegação autônoma, robótica baseada em comportamento, robô Khepera II, sistemas imunológicos artificiais, teoria da rede imunológica, algoritmos genéticos, sistemas híbridos.

Abstract

MICHELAN, R., **Evolution of immune networks for automatic coordination of elementary behaviors on robot autonomous navigation.** Campinas: FEEC, Unicamp, July 2003. Ms. Thesis – School of Electrical and Computer Engineering, State University of Campinas, 122 p.

The design of an autonomous navigation system for mobile robots, with simultaneous objectives to be satisfied, as garbage collection with maintenance of integrity, requires refined coordination mechanisms to deal with modules of elementary behavior. Models of artificial immune networks can then be applied to produce a controller based on dynamic mapping. The antibodies of the immune network are responsible for the modules of elementary behavior, in the form of ⟨condition⟩-⟨action⟩ rules, and the connections are responsible for the mechanisms of stimulation and suppression of antibodies. The network will always start an immune response when antigens are presented. These antigens represent the current output of the robot sensors. The network dynamics is based on the levels of antibody concentration, provided by interaction among antibodies, and among antibodies and antigens. Based on its concentration level, an antibody is chosen to define the robot action. An evolutionary process is then used to define the connection pattern of the immune network, from a population of candidate networks, capable of maximizing the objectives during navigation. As a consequence, a hybrid system is conceived, with an immune network implementing a dynamic process of decision-making, and an evolutionary algorithm defining the network structure. To be able to evaluate the controllers (immune networks) at each iteration of the evolutionary process, a virtual environment was developed for computer simulation, based on the characteristics of the navigation problem. The immune networks obtained by evolution were analyzed and tested in new situations and presented coordination capability in simple and complex tasks. The preliminary experiments on a real Khepera II robot indicated the efficacy of the navigation tool.

Keywords: robot autonomous navigation, behavior based robotics, Khepera II robot, artificial immune systems, immune network theory, genetic algorithms, hybrid systems.

Índice

Capítulo 1	1
Introdução e Posicionamento da Pesquisa	1
1.1 Uma solução híbrida para a navegação autônoma de robôs móveis	1
1.2 Simulação computacional e implementação em robôs reais	2
1.3 Navegação autônoma e computação evolutiva	3
1.4 Uma abordagem imunológica para navegação autônoma	4
1.5 Relações com a robótica baseada em aprendizado	6
1.6 Relações com a robótica baseada em comportamento	7
1.6.1 A partição em comportamentos elementares	7
1.6.2 Definição automática do módulo de coordenação	8
1.7 Avaliação de desempenho a partir do comportamento global	9
1.8 Relações com vida artificial	9
1.9 Relações com outras formas de estudar comportamento	10
1.10 O poder da interação com o ambiente	11
1.11 Organização do texto	12
Capítulo 2	13
Robótica Baseada em Comportamento	13
2.1 Autonomia	13
2.2 Comportamento	15
2.3 Sistemas Inteligentes	16
2.3.1 A Robótica e a Inteligência Artificial (IA)	17
2.4 Robótica Baseada em Comportamento (RBC)	18
2.4.1 Paradigmas de um sistema baseado em comportamento	19
2.4.2 Mecanismos de Coordenação	20
2.4.3 Agrupamentos Comportamentais	22
2.4.4 Características desejáveis em sistemas RBC	23
Capítulo 3	25
Sistemas Imunológicos Artificiais e Algoritmos Evolutivos	25
3.1 Auto-reconhecimento e auto-afirmação	25

3.2 Fundamentos de Imunologia	27
3.3 Teoria da Rede Imunológica	30
3.3.1 A Inspiração.....	30
3.3.2 A Ferramenta	31
3.3.3 O Modelo de Rede Imunológica Utilizado.....	33
3.4 Aspectos de Representação e Implementação Computacional.....	35
3.5 Algoritmos evolutivos	36
3.5.1 Tipos de algoritmos evolutivos	37
3.5.2 Pressão seletiva e manutenção de diversidade	37
3.5.3 Codificação.....	38
3.5.4 Operadores genéticos.....	39
3.5.5 Função de Avaliação	40
3.5.6 O processo de seleção.....	40
3.5.7 Algoritmo evolutivo básico	41
Capítulo 4	43
Uma Abordagem Imuno-Genética para Navegação Autônoma	43
4.1 Motivação para uma abordagem híbrida	43
4.2 Uma rede imunológica para navegação autônoma	45
4.2.1 Abordagens similares na literatura	46
4.2.2 Origem e motivação do problema de navegação.....	47
4.2.3 Objetivos da navegação.....	47
4.2.4 Estrutura do modelo de rede imunológica.....	48
4.2.5 Antígenos e Anticorpos	49
4.2.6 Rede Imunológica: Mecanismo de Coordenação	52
4.3 Um mecanismo de adaptação para a rede imunológica.....	54
4.3.1 A representação	56
4.3.2 A função de <i>fitness</i>	57
4.3.3 Operadores Genéticos.....	58
4.4 Ambiente de simulação	60
4.5 Arquitetura do robô	61
Capítulo 5	63

Resultados e Discussões	63
5.1 Evolução via Crossover e Mutação	64
5.2 Evolução via mutação	67
5.2.1 Evolução via mutação: tarefa de desvio de obstáculos	68
5.2.2 Evolução via mutação: tarefa de desvio de obstáculos e captura de alvo	71
5.2.3 Evolução via mutação: tarefa de coleta de lixo	77
5.3 Experimentos Práticos	82
5.3.1 Navegar pelo Corredor	83
5.3.2 Navegar ao redor do obstáculo	84
5.3.3 Dilema do “beco”	86
5.3.4 Acompanhamento da concentração	87
5.4 Considerações	95
Capítulo 6	97
Considerações finais	97
6.1 Múltiplos objetivos, dinâmica e sistemas híbridos	97
6.2 Principais contribuições	99
6.3 Perspectivas futuras	100
Apêndice A	103
Aspectos técnicos do ambiente de simulação	103
A.1 Introdução	103
A.2 Apresentação do Ambiente	104
A.3 Linguagem de Programação e Estrutura do Código Fonte	104
A.4 Modelo Computacional do Ambiente	104
A.5 Modelo Computacional do Robô	105
A.6 Execução	107
A.7 Considerações finais	108
Apêndice B	109
Aspectos técnicos do robô Khepera	109
B.1 Apresentação do Khepera	109
B.2 Características	110
B.3 Aspectos de comunicação e controle	111

B.4 Considerações Finais	112
Referências bibliográficas _____	115

Índice de Algoritmos

Algoritmo 3.1 – Algoritmo Evolutivo Básico. _____	41
---	----

Índice de Tabelas

Tabela 4.1 – Correlação entre um organismo biológico e um robô. _____	45
Tabela 4.2 – Decomposição em condições específicas. _____	49
Tabela 4.3 – Possibilidades de escolha da ação a ser associada a cada condição. _____	50

Índice de Figuras

Figura 2.1 – O robô recebe do ambiente as informações através dos sensores e utiliza seu conhecimento sobre a situação para atuar adequadamente através de seus atuadores. _____	14
Figura 2.2 – O esquema de diferenciação entre os tipos de sistema de controle. O lado deliberativo está intimamente ligado aos sistemas ditos inteligentes, que exibem capacidade de adaptação, enquanto o reativo aos sistemas de controle com ou sem inteligência, mas sem capacidade de adaptação. (Adaptado de ARKIN (1998)) _____	17
Figura 2.3 – Exemplo do fluxo de informação em sistemas em que a coordenação de comportamentos elementares pode conduzir a comportamentos mais complexos. _____	19
Figura 2.4 – O mecanismo de cooperação agrega os comportamentos mais estimulados formando uma única ação que incidirá nos atuadores do robô. _____	20
Figura 2.5 – O robô está com o nível de energia baixa e deverá recarregar as baterias antes de prosseguir, apesar de seu objetivo final ser o alvo. Esse comportamento indica que se manter operante é mais prioritário que o alcance do alvo. _____	21
Figura 2.6 – O sistema proposto por MAES (1990) faz a competição entre os comportamentos distribuídos sem hierarquia. O mais ativado é escolhido para agir. _____	21
Figura 2.7 – O sistema proposto por ROSENBLATT & PAYTON (1989) utiliza duas etapas. Na primeira os comportamentos votam nas ações a que estão relacionados. Em seguida, a ação mais votada é escolhida para atuar. _____	22
Figura 2.8 – Os grupos de comportamento permitem maior nível de estruturação na concepção do sistema. No exemplo, o grupo <desviar de obstáculos> é composto por comportamentos elementares como, por exemplo, se não houver obstáculos siga em frente. _____	23

Figura 2.9 – Agrupamento de comportamentos elementares. Um mecanismo de coordenação, seja por cooperação ou competição, deve poder atuar em dois níveis: intergrupos e intragrupos. _____	23
Figura 3.1 – O processo de reconhecimento antigênico pelos linfócitos-B. _____	28
Figura 3.2 – Esquema simplificado da atuação do sistema imunológico. Adaptado de DE CASTRO (2001). _____	30
Figura 3.3 – Observação do início do processo de fagocitose. Os macrófagos englobam o agente que é destruído através de processos físico-químicos. _____	32
Figura 3.4 – Exemplo pictórico de como se formam as redes imunológicas. Nota-se as conexões entre os paratopos de algumas células-B e os idiotopos de outras, assim como o reconhecimento do antígeno. _____	32
Figura 3.5 – Visão pictórica de mecanismos de estimulação e supressão. _____	33
Figura 3.6 – Exemplos de codificação binária, inteira e real. _____	38
Figura 3.7 – Operador crossover de um ponto ou simples. _____	39
Figura 3.8 – Operador de mutação alterando o valor do gene escolhido. _____	40
Figura 4.1 – A estrutura do anticorpo é composta pela regra <condição>-<ação> que corresponde à parte denominada paratopo e por suas conexões ou idiotopos. _____	48
Figura 4.2 – As codificações dos antígenos e dos anticorpos são arranjos binários. O antígeno tem sua estrutura idêntica ao primeiro trecho da estrutura do anticorpo, permitindo a comparação bit a bit para se determinar o grau de afinidade. _____	50
Figura 4.3 – Exemplo de situação em que a tomada de decisão não depende somente das condições externas, mas também do nível de energia. Os antígenos formados representam essa situação. ____	53
Figura 4.4 – Rede imunológica projetada visando a manutenção da integridade do robô. _____	53
Figura 4.5 – Algoritmo para cada passo do robô, que resulta em uma ação. _____	54
Figura 4.6 – Codificação utilizada para a estrutura da rede imunológica projetada por WATANABE <i>et al.</i> (1999). Um número 1 na <i>i</i> -ésima posição das conexões indica que aquele anticorpo estimula o <i>i</i> -ésimo anticorpo. _____	55
Figura 4.7 – Exemplo de aplicação do operador de crossover, onde os anticorpos A ₅ e A ₈ são trocados e as redes resultantes passam a ter nova configuração de conexões. Os demais anticorpos de cada rede imunológica não sofrem nenhuma alteração, razão pela qual não foram apresentados.	59
Figura 4.8 – O ambiente provê estruturas que representam lixos (círculos vermelhos), bases (retângulos azuis de cantos arredondados) e obstáculos (objetos em preto). _____	60
Figura 5.1 – Simulação com a rede proposta por WATANABE <i>et al.</i> (1999). _____	66
Figura 5.2 – Simulação com a rede evoluída utilizando os dois operadores genéticos. _____	66
Figura 5.3 – Estruturas das redes utilizadas nas simulações. A estrutura A representa a rede proposta no trabalho de WATANABE <i>et al.</i> (1999), com 100 conexões, e a estrutura B a rede evoluída pelo sistema imuno-genético, com 231 conexões. A estrutura C representa as 47 conexões existentes simultaneamente nas redes A e B, e a estrutura da figura D as 237 conexões diferentes nas duas redes. _____	67

Figura 5.4 – Ausência de coordenação nas decisões tomadas: o robô não consegue navegar sem colidir.	69
Figura 5.5 – Presença de coordenação nas decisões tomadas. Verifica-se que foram tomadas decisões adequadas durante a navegação, fazendo com que o robô se desvie dos obstáculos encontrados.	70
Figura 5.6 – Simulação de desvio de obstáculo com velocidade maior que a utilizada na Figura 5.5. O robô conseguiu navegar com apenas duas colisões.	70
Figura 5.7 – Estruturas das redes utilizadas nas simulações. As estruturas A e B representam as redes evoluídas pelo sistema imuno-genético, onde A possui 78 conexões e B 93 conexões. A estrutura C representa as 44 conexões iguais existentes nas redes A e B, e a estrutura da figura D as 83 conexões diferentes nas duas redes.	71
Figura 5.8 – A rede sem conexões não foi capaz de coordenar as ações adequadamente, o que implicou em um grande número de colisões. Verifica-se também a ausência de atração pelo alvo.	73
Figura 5.9 – O robô alcança o alvo desviando corretamente dos obstáculos.	73
Figura 5.10 – O robô colide em sua primeira tentativa de alcançar o alvo, mas é bem sucedido na segunda superando os obstáculos.	74
Figura 5.11 – O robô alcança o alvo mesmo com um maior número de obstáculos.	74
Figura 5.12 – O robô explora o ambiente, captura o alvo e depois segue explorando. O quadro à esquerda mostra apenas uma parte da trajetória completa apresentada no quadro à direita.	75
Figura 5.13 – Da mesma forma que na simulação anterior, o robô explora o ambiente, captura o alvo e, depois, segue explorando. O quadro à esquerda mostra apenas uma parte da trajetória completa apresentada no quadro à direita.	75
Figura 5.14 – O robô consegue sair do corredor sem colidir e realiza a captura do alvo.	75
Figura 5.15 – A seqüência mostra o robô em busca do alvo, que se move para evitar que o robô o alcance.	76
Figura 5.16 – Estruturas das redes utilizadas nas simulações. As estruturas A e B representam as redes evoluídas pelo sistema imuno-genético, onde A possui 149 conexões e B 158 conexões. A estrutura C representa as 73 conexões iguais existentes nas redes A e B, e a estrutura da figura D as 161 conexões diferentes nas duas redes.	77
Figura 5.17 – O robô ignora o lixo, buscando a base e recarregando suas baterias.	78
Figura 5.18 – O robô captura o lixo, buscando a base, depositando o lixo e recarregando suas baterias.	78
Figura 5.19 – O robô captura o lixo, mas não é estimulado pela base e começa a explorar o ambiente. Nessa trajetória ele se depara novamente com o lixo, mas o ignora, pois já está carregando lixo.	79
Figura 5.20 – O robô explora o ambiente mas as colisões e a falta de tempo não permitem que ele retorne à base novamente.	79
Figura 5.21 – O robô ignora o lixo, buscando a base e recarregando suas baterias tal como a rede evoluída.	80

Figura 5.23 – O robô, inicialmente não sendo estimulado o suficiente pelo lixo, permanece navegando em comportamento exploratório até o tempo se esgotar.	81
Figura 5.24 – A comparação das estruturas evidencia que há várias possibilidades de configurações para que a rede seja capaz de realizar a tarefa de coleta de lixo. A estrutura A representa a rede proposta em WATANABE <i>et al.</i> (1999), com 100 conexões e a B a rede obtida pelo processo evolutivo, com 282 conexões. As duas possuem 47 conexões iguais (estrutura C) e 288 conexões diferentes (estrutura D).	81
Figura 5.25 – A comparação das estruturas indica que há várias configurações possíveis para uma rede, dada uma mesma tarefa. A estrutura A exibe as 102 conexões iguais entre as redes das Figuras 5.3B e 5.24B. A estrutura B exibe as 309 conexões diferentes.	82
Figura 5.26 – Disposição dos sensores no robô Khepera.	83
Figura 5.27 – No experimento realizado, o robô foi forçado a tangenciar paredes e desviar do obstáculo posicionado ao final do corredor.	84
Figura 5.28 – Rodeando um obstáculo: o robô conseguiu realizar a navegação sem colisões.	85
Figura 5.29 – O robô evita o “beco”, contornando a parede e conseguindo escapar.	86
Figura 5.30 – O robô evita o “beco”, contornando a parede e conseguindo escapar.	87
Figura A.1 – Obstáculos (objetos em preto) e alvos (objetos em azul e vermelho).	105
Figura A.2 – Disposição dos sensores do modelo computacional do robô.	106
Figura A.3 – Exemplo de estrutura das informações provenientes de um sensor.	106
Figura B.1 – Robô Khepera II comparado com o tamanho de uma moeda de 2 euros.	110
Figura B.2 – Distribuição dos sensores. Seis deles estão dispostos na parte da frente e dois na parte de trás.	110
Figura B.3 – O robô se locomove através da tração nas duas rodas. Os pontos deslizantes lhe conferem o equilíbrio.	111
Figura B.4 – Software implementado para a validação do ambiente e da comunicação com o robô.	112
Figura B.5 – Acessórios do Khepera: câmera de vídeo e garra.	113

Capítulo 1

Introdução e Posicionamento da Pesquisa

Neste capítulo, serão feitos comentários sobre as áreas de pesquisa correlatas ao tema abordado nessa dissertação. A divisão em várias seções procurou introduzir os aspectos mais relevantes e que de alguma maneira serão explorados ao longo do texto.

Na Seção 1.11, é encontrada a organização do texto da dissertação, onde são apontados os assuntos a serem cobertos em cada capítulo, com o objetivo de fixar a idéia principal do trabalho.

1.1 Uma solução híbrida para a navegação autônoma de robôs móveis

O projeto de robôs móveis autônomos é uma das tarefas mais desafiadoras da engenharia de controle e automação. Isto fica evidente a partir do momento em que constata-se que os computadores e outros dispositivos eletro-eletrônicos estão presentes em boa parte das atividades do dia-a-dia. Mas esta presença constante não se verifica no caso de robôs móveis inteligentes, em nossos lares e em nossas cidades, mesmo que realizando apenas as atividades mais elementares e repetitivas.

A principal razão está na dificuldade de projeto, pois a interação do robô móvel com o ambiente pode ser vista como a interação de dois sistemas dinâmicos. Mesmo no caso de ambientes estáticos, como o robô é móvel, o ambiente varia no tempo, sob a perspectiva do robô. Com isso, o estado dos sensores do robô em um dado instante é função das propriedades do ambiente que circunda o robô e das ações prévias tomadas pelo robô.

Esta dissertação apresenta uma metodologia híbrida de geração automática de estratégias de controle para robôs autônomos, seguindo um paradigma baseado em comportamento. Com base neste paradigma, o robô será interpretado como um organismo artificial autônomo, capaz de desenvolver suas habilidades de navegação através da interação direta com o ambiente, sem intervenção humana. Dentre outras possibilidades de implementação, já foram empregadas com sucesso na literatura abordagens evolutivas

(NOLFI & FLOREANO, 2000; DORIGO & COLOMBETTI, 1997; FLOREANO & MONDADA, 1995) e abordagens baseadas em sistemas imunológicos artificiais, particularmente recorrendo à teoria da rede imunológica (ISHIGURO *et al.*, 1996; WATANABE *et al.*, 1999). A exploração conjunta de aspectos complementares da computação evolutiva e de redes imunológicas já havia sido realizada em WATANABE *et al.* (1998), mas o tratamento a ser apresentado neste trabalho vai requerer a síntese de redes imunológicas com um número bem superior de nós e conexões, pois enquanto WATANABE *et al.* (1998) operam com um sistema de tomada de decisão de alto nível, aqui é adotado baixo nível, de modo que as ações mais elementares do robô devam ser propostas a partir da atuação da rede imunológica.

Será considerado um problema específico de navegação multiobjetivo já proposto na literatura (PFEIFER & SCHEIER, 1999), sendo que sua escolha se deve ao fato dele ser suficientemente complexo a ponto de justificar o refinamento das abordagens de controle a serem propostas. Um robô deve navegar em um ambiente que contém obstáculos, depósitos de lixo, dotados de uma base de abastecimento de energia para o robô, e fontes de lixo a ser coletado. A tarefa do robô é caminhar pelo ambiente sem sofrer colisões, encontrar fontes de lixo, coletar o lixo e o conduzir até o depósito, isso tudo mantendo um nível mínimo de energia interna.

Existem três fontes de consumo da energia interna do robô: o simples ato de se movimentar, as colisões e o transporte de lixo. Sendo assim, as decisões do robô a cada momento não dependem apenas das informações sensoriais, mas também de seu nível interno de energia. Este quadro conduz a situações de navegação que impedem o atendimento simultâneo dos objetivos. Um exemplo é quando o robô se encontra próximo a uma fonte de lixo e está com um nível baixo de energia. Neste caso, ele deve caminhar imediatamente em direção a um depósito, mesmo sem transportar lixo algum, de modo a repor sua carga de energia interna. Uma vez repostas a carga de energia interna, retoma-se a tarefa de localização das fontes de lixo, transporte do lixo e navegação sem colisões.

Visando resolver este problema de navegação multiobjetivo, o propósito deste trabalho é utilizar a rede imunológica e sua dinâmica de regulação como uma estrutura de representação e coordenação de módulos elementares de comportamento de navegação. Fica a cargo do projetista a definição dos comportamentos elementares, sendo que a coordenação será sintetizada a partir de mecanismos evolutivos que vão atuar na definição das conexões da rede imunológica.

1.2 Simulação computacional e implementação em robôs reais

O processo de síntese do controlador e os experimentos de navegação a serem descritos ao longo deste texto foram realizados empregando um ambiente de simulação computacional, desenvolvido especificamente para o tipo de investigação científica pretendida (veja Apêndice A).

No entanto, com o advento de robôs em miniatura e de fácil manipulação, experimentos refinados de navegação passaram a ser acessíveis a pesquisadores com diferentes formações e mesmo sem o domínio da eletrônica e dos recursos de programação

em baixo nível, que normalmente eram requisitados dos grupos de pesquisa em robótica autônoma.

A partir de então, as aplicações e metodologias de análise e síntese têm se diversificado de forma significativa. Foram marcantes os experimentos evolutivos para robótica autônoma realizados nos anos de 1992 e 1993 por três grupos de pesquisa dos seguintes centros: Swiss Federal Institute of Technology, em Lausanne (Suíça), University of Sussex, em Brighton (Reino Unido), e University of Southern California, na California (Estados Unidos).

Os progressos obtidos por esses grupos ajudaram a consolidar o Khepera (K-TEAM S.A., 2003) como um robô em miniatura adequado para experimentos acadêmicos em ambientes reais de navegação. Sendo assim, alguns experimentos preliminares e parciais empregando um robô Khepera II também fazem parte dos resultados deste trabalho, visando apontar a validade da proposta e abrindo perspectivas para trabalhos futuros.

O emprego de robôs reais permite que uma quantidade muito grande de fatores relevantes ao experimento seja considerada, visto que todas as propriedades físicas do robô e do ambiente irão efetivamente participar do experimento, além do emprego de sensores e atuadores realísticos. Por exemplo: atrito, inércia, luz ambiental, ruído, etc.

A existência de um ambiente de simulação computacional e também o desenvolvimento de técnicas de laboratório para implementação das estratégias de controle em robôs reais representam resultados importantes deste trabalho de pesquisa, além das extensões relativas à proposta híbrida para o projeto do controlador, a qual é baseada em técnicas de computação evolutiva aplicadas à síntese de uma rede imunológica artificial.

1.3 Navegação autônoma e computação evolutiva

Este trabalho de pesquisa pode ser classificado junto ao ramo da robótica evolutiva, cujo início está vinculado ao trabalho de CLIFF *et al.* (1993). No entanto, o emprego de algoritmos evolutivos na síntese de controladores para robôs autônomos teve início no final da década de 1981-1990. Anterior a essas iniciativas, todo robô autônomo requeria esforços intensos de modelagem e programação explícita de todos os módulos constituintes da unidade de controle (NOLFI & FLOREANO, 2000).

A idéia básica por trás de uma abordagem evolutiva para navegação autônoma pode ser apresentada na forma resumida a seguir.

O sistema de controle como um todo, ou parte dele, possivelmente envolvendo também alguns aspectos de hardware, deve poder ser descrito na forma de uma lista de atributos. Sendo assim, cada proposta de controlador e de hardware do robô terá associada a si um cromossomo, ou seja, uma representação genética na forma de uma lista de atributos. Uma população inicial de cromossomos, cada um deles codificando uma proposta de software+hardware para navegação autônoma, é criada aleatoriamente. Em seguida, dotados de sistemas de controle implementados com a codificação de cada um dos cromossomos, robôs vão ser deixados no ambiente, livres para agir autonomamente. O desempenho de cada robô será então avaliado, de preferência de forma automática, com

base na capacidade de atender a um conjunto de tarefas de navegação. A medida de avaliação pode então ser interpretada como um índice numérico de desempenho na execução de tarefas de navegação autônoma.

Seguindo a lei da seleção natural de Darwin, descrita na Seção 3.5, os sistemas de controle que levarem os robôs a apresentarem melhor desempenho terão uma maior chance de gerar descendentes, os quais irão compor a próxima geração de propostas de sistemas de controle. Haverá basicamente três processos envolvidos: hereditariedade, ou seja, os cromossomos-filhos deverão apresentar muitos atributos similares aos atributos de seus cromossomos-pais; combinação de atributos, através da operação de crossover; e perturbações aleatórias, através da operação de mutação. Este processo de seleção e reprodução com combinação e variação deve ser repetido por algumas gerações, até que se obtenha um comportamento de navegação desejado, ou seja, um sistema de controle capaz de maximizar o atendimento dos múltiplos objetivos da navegação.

Em virtude do tempo necessário para se avaliar cada cromossomo da população (é preciso que o robô seja exposto a situações de navegação capazes de indicar a competência do controlador em atender aos requisitos de navegação), foi utilizado neste trabalho o ambiente de simulação computacional para se implementar o processo evolutivo. O controlador já evoluído é então empregado em atividades reais de navegação envolvendo o robô Khepera II.

1.4 Uma abordagem imunológica para navegação autônoma

A partir de fundamentos e conceitos de imunologia teórica e empírica, foram propostos na literatura sistemas computacionais que apresentam algumas dentre as propriedades marcantes dos sistemas imunológicos, particularmente o sistema imunológico dos mamíferos (JANEWAY *et al.*, 1999). Estes sistemas computacionais, geralmente empregados na solução de problemas envolvendo reconhecimento de padrões, cognição e otimização (FARMER *et al.*, 1986), foram denominados sistemas imunológicos artificiais (DASGUPTA, 1999; DE CASTRO & TIMMIS, 2002). Eles podem ser concebidos empregando uma formalização que envolve três estágios (DE CASTRO & TIMMIS, 2002):

1. um esquema de representação para os componentes do sistema;
2. um conjunto de mecanismos para avaliar as interações dos elementos do sistema, entre si e com o mundo exterior;
3. um conjunto de procedimentos de adaptação.

A teoria da rede imunológica (JERNE, 1974) é uma das vertentes dos sistemas imunológicos artificiais, pois os nós da rede têm associados a si uma representação, normalmente na forma de um vetor de atributos, o nível de interação dos nós é determinado por uma medida de distância entre os vetores de atributos, e o ajuste das concentrações de cada nó segue um sistema de equações diferenciais acopladas.

Basicamente, a teoria da rede imunológica propõe que o sistema imunológico é composto por um conjunto de células e moléculas que interagem dinamicamente através de

suas propriedades físico-químicas. A dinâmica da rede define os mecanismos regulatórios que determinam as concentrações de cada componente do sistema. A presença de um antígeno externo é, então, vista como uma perturbação localizada em algum ponto ou conjunto de pontos da rede imunológica, levando a uma reação do sistema no sentido de buscar um outro ponto de equilíbrio para a rede.

Cada nó da rede corresponde a um comportamento elementar de navegação na forma de uma regra <condição>-<ação>, e será associado a um anticorpo da rede imunológica. Um antígeno vai corresponder ao estado dos sensores do robô, contendo informações do ambiente externo e também de propriedades internas do robô. A dinâmica da rede imunológica vai indicar a variação no nível de concentração de cada anticorpo, o qual é função das interações mútuas com os demais anticorpos e antígenos. Estas interações são definidas pelo padrão de conexão ou estrutura da rede imunológica. Os anticorpos que se destacarem em termos de nível de concentração irão competir para indicar a ação a ser adotada pelo robô.

A teoria da rede imunológica já foi empregada na tarefa de navegação autônoma (ISHIGURO *et al.*,1996; WATANABE *et al.*,1999). A grande diferença entre as propostas de ISHIGURO *et al.* (1996) e WATANABE *et al.* (1999) frente a presente proposta está no nível de automatização de projeto. Nestas duas implementações, tanto os comportamentos elementares como o padrão de conexão da rede imunológica foram definidos pelo projetista, ou de forma arbitrária ou então recorrendo a uma metodologia construtiva responsável pela introdução um-a-um de novos componentes na rede imunológica. Aqui, apenas os comportamentos elementares serão definidos pelo projetista, ou seja, os nós da rede imunológica. O padrão de conexão da rede será definido automaticamente a partir de um mecanismo evolutivo de busca.

WATANABE *et al.* (1998) representa a proposta da literatura que mais se assemelha à presente proposta. É empregado um algoritmo evolutivo tanto para especificar nós como conexões de uma rede imunológica responsável pelo processo de tomada de decisões do robô. A principal diferença reside no fato de que WATANABE *et al.* (1998) emprega uma rede imunológica para a tomada de decisões de alto nível, do tipo <Dirija-se à estação de recarga>. Por outro lado, a presente proposta emprega um mecanismo de baixo nível, com decisões do tipo <Vire à direita>. Na prática, uma longa e diversificada série de ações de baixo nível pode ser necessária para atender a uma única ação de alto nível. Essa diferença de concepção tem impactos significativos na dimensão e na estrutura da rede imunológica em cada caso. WATANABE *et al.* (1998) convergem para redes imunológicas com 5 nós e menos de 10 conexões, dimensões essas incapazes de atender a demanda representada por controladores que atuam com decisões de baixo nível. Por outro lado, são praticamente invariantes os comportamentos necessários de baixo nível que permitem o cumprimento de tarefas sofisticadas de alto nível. Logo, ao trabalhar com decisões de baixo nível, é possível fixar os nós da rede imunológica (diretamente associados a cada decisão de baixo nível) e evoluir apenas as conexões.

1.5 Relações com a robótica baseada em aprendizado

Quando se empregam técnicas de aprendizado na síntese do controlador, este normalmente estará baseado em uma rede neural artificial (BARTO *et al.*, 1983), a qual irá tomar um conjunto de amostras de comportamento desejado de entrada-saída (certamente incapazes de cobrir todas as situações de navegação possíveis) para guiar o ajuste de seus parâmetros. A capacidade de generalização do controlador resultante, ou seja, sua habilidade de tomar decisões corretas frente a situações nunca antes experimentadas (embora estas situações devam apresentar algum grau de similaridade com um subconjunto das amostras de treinamento) é o fundamento que sustenta esta abordagem.

Repare que, em lugar de se sintetizar o controlador através de estratégias explícitas de projeto dos componentes do controlador, como será visto na seção que trata de robótica baseada em comportamento, aqui se busca alcançar um determinado comportamento de entrada-saída sem a preocupação de interpretar os mecanismos envolvidos na síntese. Em outras palavras, desde que haja meios para se realizar o ajuste automático de parâmetros que levem o controlador a se comportar da forma desejada, o domínio dos processos internos responsáveis pelo mapeamento de entrada-saída desejado deixa de ser relevante.

Em relação à abordagem desta dissertação, há três diferenças marcantes. Primeiramente, por se tratar de uma abordagem evolutiva, a estratégia a ser empregada requer muito menos supervisão ou realimentação a respeito do que deve ser feito a cada instante do processo de navegação, quando comparada ao treinamento de um rede neural artificial. Também há muito menos restrições a respeito de que tipo de aspecto de projeto pode ser ajustado automaticamente, sendo que este ajuste pode envolver a própria estrutura do controlador, além de outros parâmetros de sintonia fina. Finalmente, embora o controlador a ser apresentado também seja uma rede com interconexão de suas unidades básicas, estas unidades básicas ou nós da rede são interpretáveis e serão denominados de módulos de comportamento elementar.

Repare que aqui não se está comparando, diretamente e de forma genérica, aprendizado com evolução, e sim dois processos de síntese do controlador: um baseado em aprendizado e o outro baseado em mecanismos evolutivos. Embora não seja considerada neste trabalho, a aplicação conjunta de aprendizado e evolução em navegação autônoma também é possível, tendo sido realizada com sucesso por ACKLEY & LITTMAN (1992).

Comparações também já foram realizadas entre estratégias modulares e não-modulares para o projeto do controlador em navegação autônoma, particularmente no contexto de redes neurais artificiais (MONDADA & FLOREANO, 1995). A modularização, quando possível e realizada com critério, pode levar a desempenhos superiores, principalmente quando a complexidade da tarefa de navegação aumenta (NOLFI & FLOREANO, 2000). É aqui que se encontra a principal equivalência entre a robótica baseada em aprendizado e a proposta desta dissertação.

1.6 Relações com a robótica baseada em comportamento

A robótica baseada em comportamento visa fornecer ao robô algumas habilidades comportamentais elementares, sendo que o comportamento global do robô emerge da dinâmica de interação do robô (com seus comportamentos elementares) e o ambiente de navegação (BROOKS, 1986, 1999; ARKIN, 1998).

Como os comportamentos elementares são implementados em módulos distintos, é necessário o estabelecimento de mecanismos de coordenação para se determinar a importância relativa de cada comportamento elementar em um instante específico do processo de navegação. Se apenas um dos comportamentos elementares afeta a tomada de decisão do robô a cada momento, então diz-se que a coordenação é de natureza competitiva. Por outro lado, se a tomada de decisão do robô é uma composição de diferentes comportamentos elementares, sendo uma única proposta de composição para cada momento, então diz-se que a coordenação é cooperativa (NOLFI & FLOREANO, 2000).

Fica evidente aqui que o ambiente exerce um papel de grande relevância, pois ele interfere diretamente na definição do papel de cada comportamento elementar ao longo da navegação. Além disso, projetos de robótica baseada em comportamento empregam normalmente estratégias construtivas, com um aumento continuado no número de comportamentos elementares durante os experimentos, os quais se baseiam em tentativa e erro. Embora não seja construtiva, uma metodologia empregando tentativa e erro também estará presente na abordagem de síntese a ser apresentada neste trabalho, desde que o processo de busca realizado pelo algoritmo evolutivo possa ser interpretado como uma busca exploratória imediatamente seguida da avaliação de erros e acertos obtidos.

Uma possível dificuldade da robótica baseada em comportamento está na necessidade de particionar a atividade de navegação em um determinado número de comportamentos elementares, a serem definidos de forma intuitiva pelo projetista, e também na requisição de um módulo de coordenação. Esta dissertação também parte de um determinado número de módulos de comportamento elementar. No entanto, aqui se procura definir automaticamente o módulo de coordenação, empregando estratégias de sistemas imunológicos artificiais e computação evolutiva. Faz parte das perspectivas futuras a automatização do processo de definição dos módulos de comportamento elementar.

1.6.1 A partição em comportamentos elementares

Para dividir o comportamento em módulos elementares, é necessário um procedimento de descrição do comportamento. Existem basicamente duas perspectivas para se descrever comportamento (ARKIN, 1998): uma descrição do ponto de vista do observador (distal) e uma descrição do ponto de vista do robô (proximal). A descrição distal vai se fundamentar na interação dinâmica do robô com o ambiente de navegação, enquanto que a descrição proximal vai enfatizar diretamente as reações do robô frente a diferentes estados dos sensores.

O estado atual dos sensores é definido pelo ambiente e vai influenciar na determinação das reações do robô. Essas reações, por sua vez, modificam o ambiente ou

então a posição relativa do robô no ambiente, influenciando assim na determinação do padrão de variação do estado dos sensores.

Essas relações de dependência explicam por que é tão difícil sintetizar a atividade de navegação a partir da coordenação de módulos de comportamento elementar. É difícil prever, por exemplo, que tipo de comportamento global vai resultar a partir da atuação coordenada de um determinado número de comportamentos elementares.

A estratégia convencional para se buscar uma solução para o problema de navegação é contar com o auxílio do computador na divisão do problema em uma lista de sub-problemas mais simples. Uma divisão clássica é aquela que considera como módulos constituintes do processo de navegação: percepção, planejamento e ação. No entanto, esta divisão sofreu muitas críticas (BROOKS, 1986) e propostas alternativas estão baseadas apenas na definição de módulos de comportamento elementar, a serem devidamente coordenados.

Numa descrição distal, o projetista pode definir comportamentos elementares de navegação, como: detectar objetos, detectar alvos, evitar obstáculos, aproximar-se de alvos, explorar o espaço. No entanto, não há procedimentos sistemáticos para que o projetista possa determinar como deve ser estruturada a unidade de controle de modo a conduzir a uma coordenação efetiva desses comportamentos elementares, que levem a um desejado comportamento global do processo de navegação. Deve-se então recorrer a técnicas de busca, dentre as quais se destaca a computação evolutiva.

WATANABE *et al.* (1998) empregaram apenas a descrição distal, enquanto que aqui serão consideradas tanto a descrição distal quanto a proximal.

1.6.2 Definição automática do módulo de coordenação

Dada a inexistência de um procedimento sistemático de projeto da etapa de coordenação, esta dissertação propõe a automatização completa da tarefa de coordenação via um procedimento de busca evolutiva de redes imunológicas que vão implementar a coordenação dos módulos elementares. Sendo assim, é deixada ao projetista apenas a definição dos módulos de comportamento elementar.

De forma mais específica, o módulo de coordenação será um resultado de um processo de auto-organização em uma rede imunológica que implementa a interação de comportamentos elementares previamente definidos. Em outras palavras, a organização global do processo de coordenação será fornecida por um processo evolutivo que atuará na definição da estrutura da rede imunológica, sendo que, por sua vez, o processo evolutivo será guiado pelo resultado das avaliações vinculadas às ações do robô no ambiente.

Vale salientar que existem dois níveis de interação: dos comportamentos elementares; e do robô com o seu ambiente de navegação. Um outro aspecto de grande relevância está no fato de que o mecanismo de coordenação vai recorrer mais a aspectos da descrição proximal do que da descrição distal, no sentido de que a coordenação será função de aspectos sensorio-motores. Logo, a descrição distal influi diretamente na definição dos

módulos de comportamento elementares, enquanto que a descrição proximal é empregada na síntese dos mecanismos de coordenação.

1.7 Avaliação de desempenho a partir do comportamento global

A avaliação do controlador será realizada apenas em função da competência do robô em executar suas tarefas de navegação. Nenhum aspecto interno do controlador será levado em conta, como por exemplo interpretabilidade dos módulos que compõem o controlador.

Como a síntese do controlador vai empregar técnicas de computação evolutiva, a probabilidade de que os indivíduos das gerações iniciais sejam capazes de executar, ao menos parcialmente, as tarefas de navegação é tão menor quanto mais complexas forem estas tarefas. Se todos os indivíduos mostrarem-se igualmente incompetentes para atender os objetivos da navegação, então todos receberão a mesma avaliação e a evolução não poderá operar de forma adequada.

Sendo assim, pode-se ampliar o nível de supervisão, ou seja, avaliar não apenas o sucesso na execução das tarefas completas, mas dividi-las em subtarefas, particionando assim o processo de avaliação. Como uma alternativa a este particionamento, pode-se também iniciar a navegação a partir de simplificações das tarefas originais de navegação e, aos poucos, ir aumentando a complexidade das tarefas pela modificação do critério de avaliação. Esta última proposta foi apresentada por DORIGO & COLOMBETTI (1994; 1997), inspirada em técnicas usadas no treinamento de animais domesticados, e será adotada neste trabalho.

Em termos práticos, qualquer uma das duas propostas acima para se definir o critério de seleção do processo evolutivo, ambas a serem implementadas pelo projetista, é mais viável do que forçar o projetista a propor a estrutura do módulo de coordenação.

Repare que o critério de seleção vai sofrendo modificações ao longo do tempo, aumentando o rigor da avaliação com o aumento das gerações. Este comportamento poderia ser automatizado caso se adotasse a co-evolução como técnica de projeto, sendo que tanto a população de controladores candidatos como a população de critérios de avaliação candidatos evoluiriam com as gerações (DAWKINS & KREBS, 1979).

1.8 Relações com vida artificial

A área de pesquisa em vida artificial representa todas as iniciativas de estudar os fenômenos da vida através de sua simulação em sistemas artificiais, geralmente na forma de modelos computacionais. Como uma consequência imediata dessas iniciativas, busca-se investigar como entidades em diferentes níveis de organização interagem, mesmo que para tanto seja necessário introduzir simplificações junto aos processos envolvidos (PARISI, 1997).

Para atingir este objetivo ambicioso, a pesquisa em vida artificial recorre à teoria de sistemas dinâmicos, aos fundamentos de sistemas complexos e ao grande poder de processamento e memória hoje disponibilizados pelos computadores (LANGTON, 1995).

Todos esses aspectos também estão envolvidos no tratamento a ser desenvolvido aqui para navegação autônoma de robôs. A complexidade do sistema de controle resultante vai permitir analisá-lo em diferentes níveis, sendo que as propriedades predominantes em um nível vão resultar das inúmeras interações de elementos simples em níveis mais baixos de análise. É muito comum constatar que essas propriedades resultaram de processos de auto-organização, pois estes podem se manifestar quando estão presentes mecanismos de realimentação positiva, realimentação negativa e interação local.

Como a identidade do sistema é uma consequência global de interações locais sob regras simples e bem definidas, diz-se que ocorre a emergência de propriedades que caracterizam o sistema. Como o número e a intensidade das interações pode ser muito grande, não-linear e variante no tempo, torna-se impossível prever os efeitos globais do comportamento local. Em outras palavras, as propriedades emergentes não podem ser facilmente previstas ou inferidas a partir do conhecimento das regras que governam as interações. O contrário também não é simples de ser previsto ou inferido, ou seja, propor as regras de interação a partir da observação de uma dada propriedade global.

Esta é uma das principais motivações para a robótica evolutiva, principalmente quando ela atua na definição da quantidade e qualidade das interações dos elementos do sistema, pois a complexidade existente impede a adoção sistemática de uma abordagem construtiva, em que se inicia pelas partes para se chegar gradativamente ao todo.

Uma consequência muito positiva para a pesquisa em robótica autônoma é o fato do comportamento do robô móvel ser uma propriedade emergente da interação do robô com o ambiente. Com isso, pode-se ter robôs simples produzindo comportamentos complexos. Este aspecto será explorado no decorrer do texto, e representa uma das maiores contribuições da área de vida artificial para a navegação autônoma, em termos de paradigma de comportamento.

1.9 Relações com outras formas de estudar comportamento

O estudo de comportamento é realizado junto a disciplinas distintas, que podem ser divididas em dois grandes paradigmas de investigação:

- concentração na estrutura: biologia molecular, biologia celular, genética e neurociência;
- concentração nos processos: biologia evolutiva, ecologia, psicologia.

Psicofísica e neuropsicologia são disciplinas que, por sua vez, procuram fazer a ponte entre os dois paradigmas para o estudo do comportamento. No entanto, nos últimos anos duas áreas de pesquisa têm se destacado no tratamento conjunto de estrutura e processos:

- conexismo: propõe redes neurais artificiais como um ferramental teórico capaz de permitir o estudo tanto de aspectos comportamentais e cognitivos, por um lado, como de aspectos do próprio sistema nervoso, por outro lado (FIESLER & BEALE, 1996; CHIEL & BEER, 1997);

- cognição corporificada: trata do estudo de processos cognitivos e comportamentais em dispositivos físicos em interação com o ambiente (BROOKS, 1991; VARELA *et al.*, 1991; PFEIFER & SCHEIER, 1999). Por exemplo: robôs móveis em navegação autônoma.

Esta dissertação também procura realizar esta ponte entre estrutura e processos. Em linhas gerais, a percepção não será vista aqui como uma exposição passiva a uma cadeia de estímulos sensoriais, totalmente dissociada das reações a serem tomadas. Ao contrário, há um forte acoplamento entre percepção e ação, inclusive porque a principal contribuição está na síntese do processo de coordenação do mecanismo sensório-motor, sendo que o controlador será evoluído a partir do desempenho de navegação do robô.

O comportamento global será sempre um produto da interação do robô com o ambiente, não podendo ser resultado apenas de mecanismos internos do robô. Com isso, o objetivo último é coordenar percepção e ação na geração de comportamentos adaptativos que conduzam ao atendimento dos requisitos de navegação.

1.10 O poder da interação com o ambiente

A robótica móvel não tem explorado devidamente os efeitos da interação com o ambiente na definição do comportamento global da navegação autônoma. A primeira demonstração precisa de que um robô dotado de módulos de comportamentos elementares é capaz de produzir comportamentos globais complexos pela exploração da complexidade do ambiente foi produzida por BRAITENBERG (1984).

E a razão para esta ausência de exploração efetiva da interação com o ambiente está no fato de que comportamentos adaptativos são difíceis de serem concebidos via projeto sistemático. Para justificar esta afirmação, considere os dois aspectos sensório-motores a seguir:

- cada ação motora vai contribuir de forma incremental para o sucesso da tarefa de navegação;
- o próximo estado sensorial é função do estado sensorial anterior, da ação motora adotada e do estado atual do ambiente.

Como a seqüência de estados sensoriais permite estabelecer o grau de progresso do robô na execução de suas tarefas, então definir de forma sistemática que ação motora adotar a cada instante de modo a maximizar o atendimento dos objetivos da navegação é algo extremamente complicado, visto que cada ação motora pode apresentar conseqüências de longo prazo para a navegação e o efeito de uma ação motora será função de ações precedentes e da interação com o ambiente.

Esta linha de argumentação evidencia mais uma vez que a automatização do processo de coordenação de comportamentos elementares é uma estratégia de projeto altamente desejável na síntese de comportamentos adaptativos, pois o projetista normalmente é incapaz de atuar apropriadamente nesta fase do projeto.

1.11 Organização do texto

Os demais capítulos desta dissertação estão organizados como segue. O capítulo 2 trata de robótica baseada em comportamento, onde serão apresentadas as principais características deste tipo de paradigma. Esse capítulo tem como objetivo propor a rede imunológica artificial como mecanismo de coordenação para módulos de comportamentos elementares, identificando aspectos da rede imunológica que expressam todas as principais características do paradigma baseado em comportamento. O capítulo 3 aborda conceitos e teorias do sistema imunológico que são usualmente empregadas na proposição de ferramentas de sistemas imunológicos artificiais. É enfatizado o modelo de rede imunológica a ser utilizado no sistema de navegação, apontando seus principais componentes. Ainda no capítulo 3, o mesmo é feito no caso da computação evolutiva, sendo apresentados os conceitos que possibilitarão a sua aplicação na evolução das redes imunológicas. No capítulo 4, é apresentada a proposta desta dissertação, onde são descritos os modelos das estruturas que compõe o sistema imuno-genético utilizado para o controle de navegação autônoma. Já no capítulo 5, são feitas considerações sobre os resultados obtidos. As primeiras seções apresentam os resultados em simulação computacional (ambiente virtual) e em experimentos com um robô real (Khepera II). Por último, é exibido o acompanhamento do nível de concentração dos anticorpos durante um experimento de navegação, a fim de evidenciar a aplicabilidade da rede imunológica como módulo de coordenação, além de validar o processo evolutivo como meio de obtenção automática de redes adequadas para o problema de navegação. As conclusões e as perspectivas para trabalhos futuros são então enunciadas no capítulo 6. Além disso, no Apêndice A é descrito o ambiente de simulação computacional implementado para a determinação do controlador, e no Apêndice B são descritos os aspectos técnicos mais relevantes sobre o robô Khepera II, utilizado nos experimentos realizados.

Capítulo 2

Robótica Baseada em Comportamento

Ao longo da década de 1981-1990, a concepção de grupos de comportamentos elementares ganhou força. Basicamente, esta concepção se refere à noção de que, para todas as possíveis situações a serem enfrentadas por um robô autônomo, há um grupo de comportamentos ou um comportamento específico que permite o atendimento dos objetivos da navegação. Assim, os sistemas baseados em comportamento são compostos por módulos de comportamentos que, devidamente coordenados, produzem comportamentos emergentes complexos não projetados a priori nos módulos.

Na Seção 2.1, são apresentados aspectos para a concepção de robôs autônomos, mais especificamente as características que definem sistemas autônomos e também os aspectos que os diferenciam de sistemas automáticos.

Nas Seções 2.2 e 2.3, são levantados os caminhos pelos quais a robótica foi inspirada e o encontro com a Inteligência Artificial, que proporcionou novos meios de obter sistemas de controle que fossem capazes de realizar as tarefas desejadas.

O ponto principal desse capítulo é posicionar as redes imunológicas dentro do paradigma de robótica baseada em comportamento. Na Seção 2.4, são levantados aspectos que permitem que a rede imunológica seja vista como mecanismo de coordenação para os módulos de comportamento elementar em tarefas de navegação autônoma.

2.1 Autonomia

A concepção de sistemas de controle para navegação autônoma tem como objetivo a realização de algum tipo de tarefa como, por exemplo, desvio de obstáculos e captura de alvos. Os robôs percebem ou recebem informações do ambiente através de seus sensores e atuam nesse ambiente fazendo uso de seus atuadores (veja Figura 2.1). Para exemplificar, sensores de infravermelho e câmeras de vídeo permitem a recepção de informações do ambiente no qual o robô está inserido, e braços e rodas operam como atuadores.

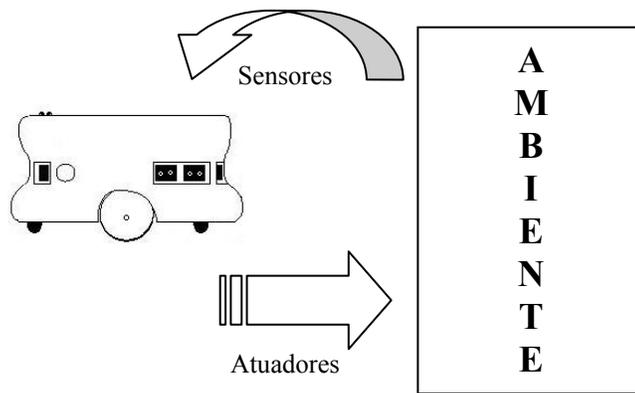


Figura 2.1 – O robô recebe do ambiente as informações através dos sensores e utiliza seu conhecimento sobre a situação para atuar adequadamente através de seus atuadores.

Definem-se três tipos de robôs. O primeiro é aquele no qual o conhecimento é embutido pelo projetista, incapaz de aprender com as experiências e estando limitado ao conhecimento a priori que lhe foi incorporado. O segundo é aquele que possui conhecimento também fornecido pelo projetista, mas diferentemente do primeiro é capaz de melhorar seu desempenho através do aprendizado, a cada nova situação. O terceiro não possui conhecimento a priori, pois foi construído para ser capaz de sozinho extrair todo o conhecimento necessário do ambiente em que está inserido, a fim de maximizar seu desempenho. O primeiro tipo está associado com controle automático, e o segundo e o terceiro tipos referem-se a controle autônomo.

O segundo tipo de robô é o mais freqüente em implementações que visam atribuir autonomia ao processo de navegação, pois permite a inclusão de conhecimento inato, a ser complementado via aprendizado.

A autonomia fornece a capacidade de tomar decisões sem a intervenção de nenhum agente externo, ou seja, baseado unicamente nas próprias experiências do robô. Para STEELS (1995), a autonomia nasce da necessidade do sistema de ser auto-suficiente. Em outras palavras, da preocupação com a própria manutenção nasce a necessidade do sistema ser autônomo.

O conceito de autonomia, no entanto, é freqüentemente confundido com automatização. Isso acontece porque esses dois tipos de sistema exibem capacidade de realizar tarefas sem a intervenção de um agente externo. Porém, a definição de autonomia é mais complexa, e pode até envolver conceitos como inteligência e consciência, embora estes não sejam propósitos desta dissertação.

Um sistema automático é um sistema auto-regulatório que utiliza a relação pré-estabelecida entre suas estruturas básicas para atuar, de maneira a realizar suas tarefas. Um sistema autônomo é auto-regulatório e autogovernado, visto que é capaz de alterar a relação entre as estruturas a fim de adequar seu comportamento, ou seja, adaptar-se às mudanças do ambiente onde o sistema se encontra (STEELS, 1995).

Porém, para que o sistema seja autônomo ele deve primeiramente ser automático. Isso porque o sistema autônomo deve ser capaz de, a partir de estruturas básicas, criar suas próprias leis (auto-regulatório). Com isso, a ação conjunta dessas estruturas pode ser coordenada para permitir a interação com o ambiente, extraindo informações e agindo para a realização da tarefa e a manutenção da própria existência. Além disso, caso ocorram mudanças no ambiente, o sistema deve ser capaz de alterar as relações entre as estruturas, o que implica em capacidade de formar e adaptar seu comportamento (auto-governado).

Outro fator importante é que essa adaptação do sistema deve ocorrer enquanto o sistema interage com o ambiente, de modo que não é possível para o sistema gastar longo tempo de processamento para escolher como e quando atuar. O sistema deve atuar constantemente, a fim de se manter operante.

Uma definição concisa de sistemas autônomos é aquela preconizada por MCFARLAND (1994):

“Um agente autônomo é autocontrolado, não estando, portanto sob o controle de um agente externo. Para ser autocontrolado, o agente deve ter conhecimento relevante e motivação. Em outras palavras, um agente autônomo deve saber o que fazer para exercer o controle no sentido de atender seus objetivos.”

Dessa maneira, conclui-se que o conceito de auto-suficiência leva o sistema à necessidade de ser autônomo. Para ser autônomo, ele deve ser auto-regulatório, para definir as próprias leis de coordenação, e auto-governado, o que o faz capaz de se adaptar às mudanças do ambiente.

2.2 Comportamento

Ao falarmos em robôs, a primeira idéia que nos vem à cabeça é aquela de um “humanóide” capaz de reproduzir ações humanas com a mesma capacidade e até melhor. Mas a robótica ainda não permite este grau de sofisticação de projeto. Assim, o que se tem hoje são robôs com alguma habilidade, mas que ainda não são tão capazes quanto os seres humanos, particularmente quando se considera um elenco amplo de cenários de atuação.

Fazendo um retorno às origens, os primeiros robôs construídos lançavam mão de algoritmos altamente desenvolvidos, mas que não conseguiam ser suficientemente abrangentes. Desde os anos 50, o homem tenta reproduzir nas máquinas comportamentos tipicamente expressos por seres humanos, com a esperança de que eles consigam executar tarefas com a precisão e eficiência humana (WIENER, 1948; WALTER, 1953).

Embora conseguissem realizar algumas tarefas os sistemas não possuíam a capacidade de adaptação necessária, provavelmente por não possuírem meios de absorver conhecimento do ambiente em que estavam imersos. Dessa maneira, com o surgimento e desenvolvimento de técnicas de inteligência artificial, os pesquisadores tiveram suas esperanças reavivadas. Metodologias de projeto baseadas em sistemas especialistas e estudos mais aprofundados em representação do conhecimento conduziram a novas possibilidades de obter sistemas adaptativos. Apesar dessas novas linhas de pesquisa, os projetos ainda estavam vinculados ao paradigma *top-down* (RUSSELL & NORVIG, 2003).

Somente anos depois é que propostas de projeto vinculadas ao paradigma *bottom-up* tomaram força e novas concepções de controladores autônomos foram desenvolvidas, principalmente baseadas em comportamento.

2.3 Sistemas Inteligentes

Uma definição formal para robôs de uso industrial é a criada pela *Robotics Industry Association* (RIA) onde o robô é considerado um “manipulador multi-função e reprogramável construído para mover materiais, peças, ferramentas, ou periféricos de movimentos programáveis, para desempenhar tarefas variadas” (JABLONSKI & POSEY, 1985). Nesta dissertação, não há uma restrição ao uso industrial, de modo que é mais adequada a definição criada por BRADY (1985) e aceita por ARKIN (1998), em que o robô é visto como uma máquina capaz de extrair informação do ambiente e usar conhecimento sobre esse mundo exterior para se mover a salvo de modo objetivo. Essa última definição vem de encontro à concepção de robô adotada nesta dissertação, e também às características de agentes autônomos descritas anteriormente.

Quando a robótica se estabeleceu como área de pesquisa, possivelmente não havia a preocupação direta com que os robôs expressassem comportamentos ditos inteligentes. Buscava-se apenas a realização de algumas tarefas elementares e repetitivas. Além disso, não havia disponibilidade de conceitos e técnicas para que algo mais sofisticado fosse buscado.

Com a evolução das pesquisas e o aparecimento de técnicas que proporcionavam cada vez mais recursos de análise e síntese de estruturas de processamento de informação, certamente auxiliada pelo crescimento na disponibilidade de recursos computacionais, a idéia de construir sistemas de controle inteligente foi ganhando força e viabilidade. Houve então a necessidade de se classificar as diversas abordagens como pertencentes ou não à vertente de sistemas inteligentes.

Considera-se um robô inteligente aquele capaz de interagir com o ambiente de modo a manter-se a salvo, mover-se de forma coordenada, construir conhecimento através da percepção, ser capaz de aprender e se adaptar ou ainda esboçar algum tipo de comportamento coletivo. Nessa direção estava WIENER (1948), que aplicou teoria de controle juntamente com ciência da informação e biologia para tentar explicar comportamentos naturais através de implementações em máquinas.

Aplicando esses princípios, WALTER (1953) criou o precursor dos robôs modernos, o qual chamou de *Tortoise*. Esse robô apresentava propriedades que estão, ainda hoje, intimamente relacionadas com a robótica, principalmente com a robótica inteligente.

Pode-se classificar estas propriedades em:

- **Parcimônia:** Reflexos simples podem servir como base para outros comportamentos mais elaborados;
- **Exploração:** O sistema nunca fica estático a não ser quando está recarregando;

- **Atração:** O sistema é motivado a se dirigir na direção de algum objeto do ambiente, ou seja, um estímulo positivo;
- **Aversão:** O sistema repele algum objeto do ambiente, ou seja, um estímulo negativo;
- **Discernimento:** Habilidade de distinguir entre comportamentos produtivos e improdutivos.

Analisando com cuidado esses princípios, observa-se que desde os primeiros modelos de sistema de controle havia a preocupação de definir comportamentos básicos dos quais outros comportamentos complexos poderiam emergir.

Outro fator que deve ser observado na modelagem de robôs é a tarefa que eles são designados a executar, uma vez que disso dependerá o tipo de locomoção e sensores a serem empregados.

Além dessas especificações, deve-se observar as diferenças entre os sistemas puramente reativos e os sistemas deliberativos (veja Figura 2.2), ressaltando assim os pontos considerados mais importantes para modelar de forma adequada o sistema de controle.

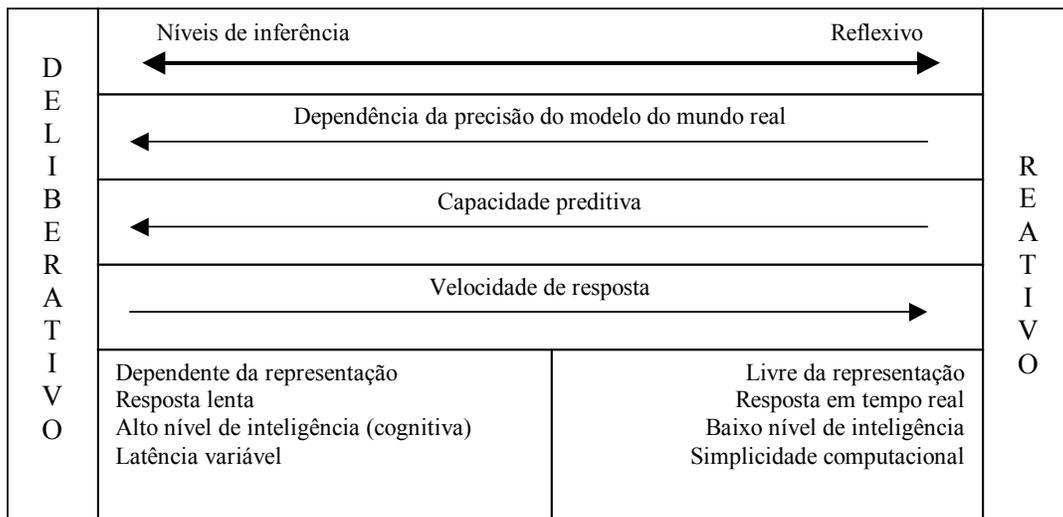


Figura 2.2 – O esquema de diferenciação entre os tipos de sistema de controle. O lado deliberativo está intimamente ligado aos sistemas ditos inteligentes, que exibem capacidade de adaptação, enquanto o reativo aos sistemas de controle com ou sem inteligência, mas sem capacidade de adaptação. (Adaptado de ARKIN (1998))

2.3.1 A Robótica e a Inteligência Artificial (IA)

Depois do nascimento da IA através da *Darmouth Summer Research Conference*, deu-se novos rumos para a pesquisa em robótica. Nessa época e até meados da década de 1981-1990, a preocupação principal foi desenvolver algoritmos com fortes raízes na

Representação do Conhecimento e Organizações Hierárquicas (*Strips*, FIKES & NILSON 1971, *ABStrips*, SACERDOTI, 1974; *Hacker*, SUSSMAN, 1975; *NOAH*, SACERDOTI, 1975) o que colocou a navegação na linha da planificação. Nessa linha de projeto, a situação corrente do robô é encarada como uma única estrutura, que não é decomponível em outras situações mais específicas. Sendo assim, esses algoritmos tentavam fazer com que os robôs de alguma forma montassem mapas dos ambientes e assim pudessem navegar de forma segura.

As principais barreiras foram dificuldade de generalização a novos contextos de aplicação e o tamanho dos sistemas desenvolvidos. Com seus conceitos de robótica baseada em comportamento, BROOKS (1987), contrariando a IA clássica, impulsionou o abandono da planificação, abrindo assim caminho para a navegação autônoma inteligente. As idéias de sensação e atuação no ambiente colaboraram para o desenvolvimento da nova linha, chamada de Robótica Baseada em Comportamento (*Behavior Based Robotics*).

2.4 Robótica Baseada em Comportamento (RBC)

Nos últimos anos, tem aumentado o interesse na área de Robótica Baseada em Comportamento. A principal motivação é encontrada nos próprios desafios que a pesquisa na área de navegação enfrenta. Como citado anteriormente, foram praticamente 30 anos de planificação durante o período inicial da pesquisa em navegação de robôs. A maioria dos métodos aplicados tentava reproduzir ou modelar o mundo real, através de mapas. Sabe-se, no entanto, que “o melhor modelo para o mundo real é ele próprio” (BROOKS, 1989). Assim, é impossível criar modelos do mundo em sua completude. Caso isso fosse factível, não haveria a possibilidade de se produzir comportamentos genuinamente emergentes, pois todos estariam modelados a priori.

Entretanto, modelos parciais com grande qualidade podem ser construídos a partir de informações simplificadas, que são extraídas na maioria das vezes através da observação do próprio sistema ou de outros mais simples.

Sob essa ótica, estão fundamentados os principais conceitos da RBC. Ela objetiva fugir da representação global do mundo real. Assim, ao supor que as ações que levam à expressão de comportamentos inteligentes são agrupamentos de comportamentos mais simples (veja Figura 2.3), é possível partir de estruturas simples e chegar a sistemas de controle altamente eficientes. Por definição, os sistemas baseados em comportamento são estruturalmente modulares, visando permitir a representação de comportamentos elementares.

Essas estruturas são apresentadas em vários trabalhos (ISHIGURO *et al.*, 1996; MAES, 1990; MILLÁN, 1996) como “*competence modules*” (módulos de competência) ou módulos elementares. Muitas são as representações adotadas para essas estruturas, algumas vezes em formato de autômatos, outras na forma de regras. O importante é que, qualquer que seja a representação, elas permitam que a informação se apresente de forma distribuída em vários comportamentos simples que, agrupados e coordenados de maneira adequada, sejam capazes de promover a emergência de comportamentos inteligentes.

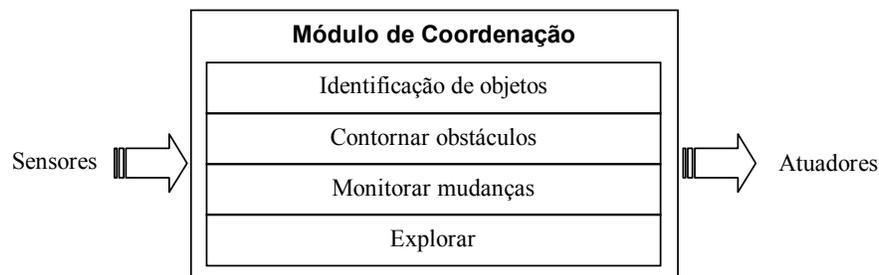


Figura 2.3 – Exemplo do fluxo de informação em sistemas em que a coordenação de comportamentos elementares pode conduzir a comportamentos mais complexos.

2.4.1 Paradigmas de um sistema baseado em comportamento

Para construir a arquitetura de um sistema baseado em comportamento, torna-se necessário recorrer a um dos paradigmas disponíveis, a fim de facilitar a elaboração dos módulos e também a maneira como será feita a coordenação entre eles. Além disso, a definição de um paradigma fornece inspiração para extensões que possam gerar melhorias de desempenho. Basicamente, distinguem-se três paradigmas de comportamento:

- ***Etológico***¹: baseado na observação de comportamentos de animais e restrito àqueles comportamentos que podem ser computacionalmente modelados e geram, de alguma maneira, padrões temporais similares àqueles observados no ambiente natural, validando assim a escolha (MCFARLAND & BOESSER, 1994);
- ***Condição-Ação***: baseado na proposição de ações elementares junto a situações específicas de navegação (DORIGO & COLOMBETTI, 1997). Para cada situação específica de navegação, supõe-se haver um comportamento local adequado no sentido de contribuir no atendimento dos objetivos da navegação. O sistema proposto nesta dissertação enquadra-se nesse paradigma e será detalhado nos Capítulos 3 e 4;
- ***Experimentação***: os comportamentos são construídos de maneira *bottom-up*, ou seja, são definidos de acordo com as situações enfrentadas pelo robô (MILLÁN, 1996; CRESTANI *et al.*, 2002).

Esses paradigmas conduzem a diferentes definições dos módulos ou comportamentos elementares. No caso específico do paradigma condição-ação, a idéia principal dos sistemas baseados em comportamento é a emergência de comportamentos complexos a partir dos mais simples. Entretanto, sabe-se que esses comportamentos são resultantes da interação do robô com o meio em que ele se encontra e da aplicação coordenada de seus módulos de comportamento elementar.

A coordenação é tão importante quanto a modelagem adequada dos comportamentos e do robô em si. É através dela que o robô realiza a escolha de qual ação tomar em cada instante. Esse assunto é amplamente discutido pelos pesquisadores da área e

¹ Etologia: estudo dos hábitos dos animais e da sua acomodação às condições do ambiente.

declarações que reforçam a tese de que os comportamentos complexos emergem da coordenação adequada dos módulos de comportamentos elementares podem ser encontradas em STEELS (1990) e BROOKS (1991).

2.4.2 Mecanismos de Coordenação

Um mecanismo de coordenação pode ser guiado pela competição ou pela cooperação dos componentes que compõe o sistema.

Como citado anteriormente, os sistemas baseados em comportamento, que são em sua essência sistemas modulares, necessitam de algum mecanismo que seja capaz de coordenar os comportamentos simples ou módulos elementares à medida que o robô interage com o ambiente, de maneira a gerar comportamentos complexos e com bom desempenho. Dentro dos procedimentos de cooperação ou competição, fica claro que para cada aplicação ou tipo de sistema existe o mecanismo mais adequado.

Em alguns modelos de sistema, os módulos ou comportamentos elementares atuam de maneira cooperativa, de modo que as decisões são tomadas agregando-se dois ou mais comportamentos mais estimulados. A decisão a ser tomada é uma composição das ações definidas em cada módulo (veja Figura 2.4).

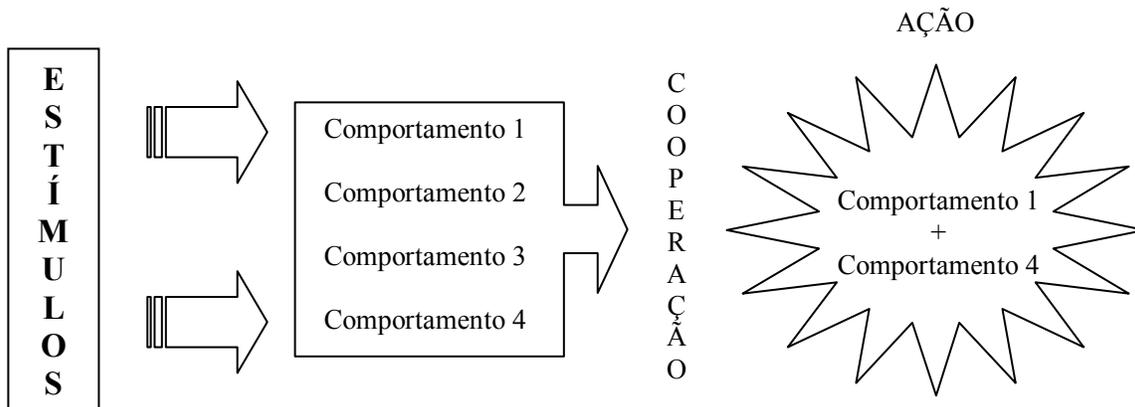


Figura 2.4 – O mecanismo de cooperação agrega os comportamentos mais estimulados formando uma única ação que incidirá nos atuadores do robô.

Dentro dessa definição, existem dois tipos de agregação. O primeiro utiliza apenas os comportamentos mais estimulados, os quais terão suas ações compostas de forma a gerar a ação final. O segundo tipo faz a composição de todos os comportamentos levando em consideração a excitação de cada um perante os estímulos recebidos.

Para uma outra classe de modelos, o mecanismo de coordenação implementado é baseado na competição entre os comportamentos simples e o mais excitado tem sua ação escolhida para alimentar o atuador. Esse tipo de coordenação pode ser aplicado basicamente por dois métodos: hierárquico e votação.

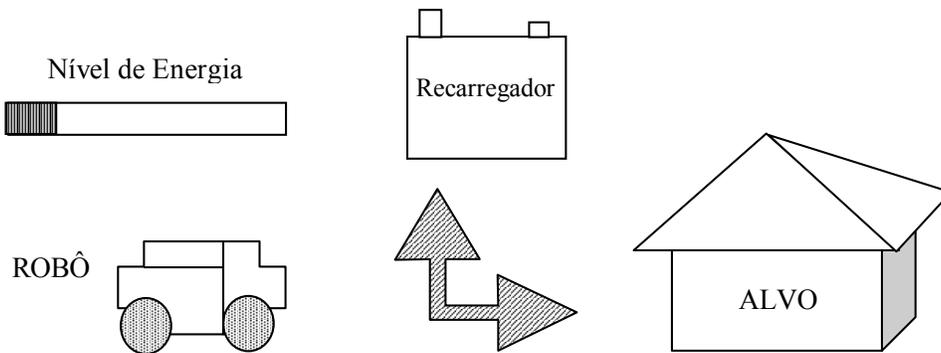


Figura 2.5 – O robô está com o nível de energia baixa e deverá recarregar as baterias antes de prosseguir, apesar de seu objetivo final ser o alvo. Esse comportamento indica que se manter operante é mais prioritário que o alcance do alvo.

Através da atribuição prévia de prioridades entre os comportamentos, o método hierárquico, ou também de rede de prioridades, escolhe o único comportamento que atuará. Pode-se ilustrar uma situação de uma navegação autônoma onde, por exemplo, o comportamento de “recarregar as baterias” seja mais prioritário que “alcançar o alvo” (veja Figura 2.5).

O método de votação leva em consideração apenas o valor ou quantidade de “votos” que um comportamento recebe durante a competição entre os comportamentos, após a entrada dos estímulos. MAES (1990) utilizou em seu trabalho um método de Ação-Seleção (veja Figura 2.6), onde os comportamentos competiam pelo controle do robô depois de receberem os estímulos ou informações provenientes dos sensores, sem que nenhuma classe hierárquica fosse pré-definida.

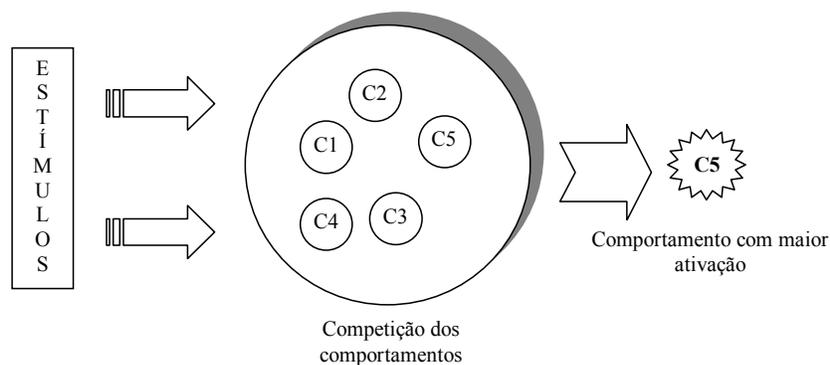


Figura 2.6 – O sistema proposto por MAES (1990) faz a competição entre os comportamentos distribuídos sem hierarquia. O mais ativado é escolhido para agir.

Em outro trabalho, ROSENBLATT & PAYTON (1989) utilizaram um sistema em que os comportamentos excitados votavam nas ações, e a ação mais votada dentre todas era a escolhida para atuar (veja Figura 2.7). Esse sistema foi implementado posteriormente em 1995 e os comportamentos eram formados por seqüências de ações. Os comportamentos foram modelados com uma estrutura que permite a identificação das ações de maneira direta e interpretável. Ex: *virar_direita*, *ir_em_frente*, *diminuir_velocidade*. Apesar do método se caracterizar por ser uma abordagem competitiva na visão das ações, a votação dos comportamentos caracteriza certo nível de cooperação.

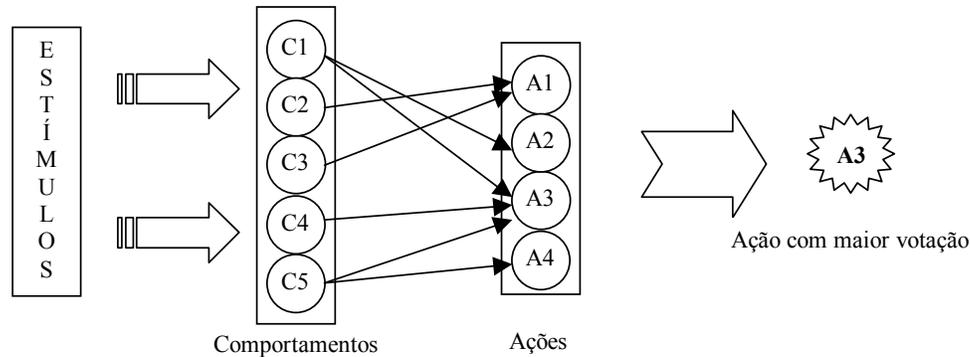


Figura 2.7 – O sistema proposto por ROSENBLATT & PAYTON (1989) utiliza duas etapas. Na primeira os comportamentos votam nas ações a que estão relacionados. Em seguida, a ação mais votada é escolhida para atuar.

Diante dessas abordagens, é possível antecipar uma associação que será tratada com maior profundidade no capítulo 4, que aborda a proposta desta dissertação. Dadas as definições acima, sabe-se agora que mecanismos de coordenação são essenciais em sistemas robóticos baseados em comportamento. Além disso, pode-se associar diretamente o uso das redes imunológicas a um tipo de mecanismo de coordenação que utiliza um método competitivo semelhante ao de ROSENBLATT & PAYTON (1989), tratando variáveis contínuas (e.g. nível de concentração de cada anticorpo) e incorporando algumas qualidades dos sistemas imunológicos artificiais, como tolerância a falhas, distribuição da informação e dinâmica regulatória.

2.4.3 Agrupamentos Comportamentais

Além dos mecanismos de coordenação e dos módulos mais básicos ou comportamentos elementares já descritos, os sistemas baseados em comportamento podem possuir estruturas que seriam, assim como o próprio sistema, descritos como agrupamentos de comportamentos (veja Figura 2.8). Esses grupos agregam tipos específicos de comportamentos vinculados a uma determinada situação representativa do processo de navegação. Por exemplo, todos os comportamentos básicos que são agrupados sob o título de <capturar alvo> ou <recarregar as baterias de um robô> ou ainda <desviar de obstáculos>, como ilustrado na Figura 2.8.

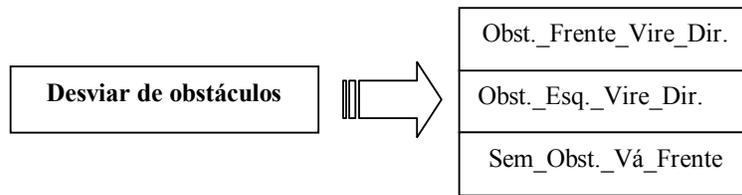


Figura 2.8 – Os grupos de comportamento permitem maior nível de estruturação na concepção do sistema. No exemplo, o grupo (desviar de obstáculos) é composto por comportamentos elementares como, por exemplo, se não houver obstáculos siga em frente.

Da mesma forma que os sistemas que utilizam comportamentos elementares sem agrupamento, os sistemas definidos por grupos de comportamentos são coordenados por mecanismos de cooperação ou competição (veja Figura 2.9).

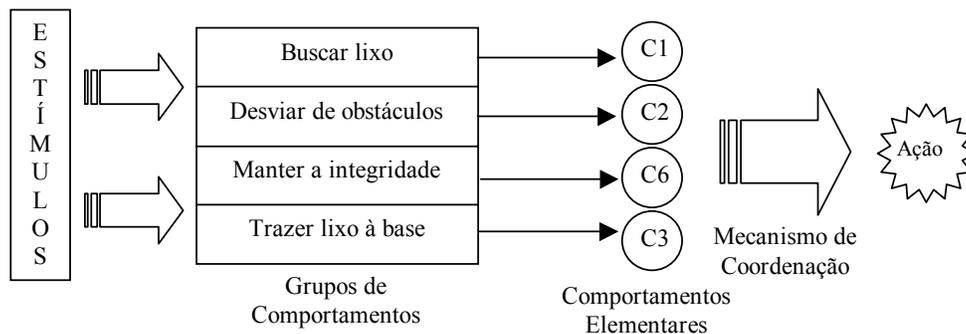


Figura 2.9 – Agrupamento de comportamentos elementares. Um mecanismo de coordenação, seja por cooperação ou competição, deve poder atuar em dois níveis: intergrupos e intragrupos.

Estruturalmente, esses grupos podem ser definidos recursivamente, o que proporciona liberdade de criação, fornecendo ao projetista a capacidade de modelar o sistema o mais próximo possível de suas inspirações, como no caso de sistemas baseados em comportamentos de animais.

2.4.4 Características desejáveis em sistemas RBC

Acima de todas as particularidades que cada modelagem é capaz de expressar, pode-se descrever algumas características que são desejáveis e que servem como critério de avaliação de sistemas que são baseados em comportamento (ARKIN, 1998).

Essas características procuram definir que tipo de arquitetura é necessária para viabilizar a aplicação da modelagem por comportamento ao problema em questão:

- **Suporte ao paralelismo:** os sistemas RBC são intrinsecamente paralelos e, por isso, a arquitetura deve prover essa característica.

- **Adequação ao hardware:** a arquitetura deve permitir adequação ao hardware em dois sentidos. Primeiro, a aplicação a sistemas robóticos reais, sendo compatível e factível no que diz respeito a sensores e atuadores. O segundo, relacionado à transferência do controle para o robô, não sendo mais necessário o controle via uma máquina acoplada.
- **Adaptabilidade ao meio:** o sistema deve prover ao robô a capacidade de se adaptar ao ambiente e realizar a tarefa para a qual foi designado.
- **Suporte à modularidade:** a arquitetura deve ser capaz de suportar as definições dos comportamentos modulares em nível razoável de abstração.
- **Robustez:** a arquitetura deve ser robusta o suficiente para encontrar meios de corrigir ou atenuar problemas como ruídos nos sensores, ou mesmo falha de um atuador.
- **Aplicabilidade:** deve-se definir se a arquitetura tem seu ponto central em estudos sobre aspectos psicológicos ou cognitivos, ou na execução de tarefas básicas.
- **Flexibilidade:** a arquitetura provê condições de alteração no sistema de controle em tempo de execução, ou seja, ela possui bons níveis de flexibilidade e adaptação, ou poder de aprendizagem.
- **Desempenho:** a arquitetura deve permitir que o robô realize a tarefa a que foi designado, com desempenho satisfatório.

Todos esses itens devem ser considerados quando da síntese de um sistema de controle para navegação autônoma, implementado com base nas propriedades de redes imunológicas artificiais. O ponto mais desafiador da abordagem encontra-se no fato de que as redes imunológicas possuem dinâmica e metadinâmica altamente complexas. Devido a isso, a criação dessas redes, mesmo por um especialista, é uma tarefa muito custosa. WATANABE *et al.* (1999) chegaram a implementar um algoritmo construtivo, através de mecanismos de ajuste e inovação, para geração de redes imunológicas dedicadas a controle autônomo. Visto que não é possível generalizar prontamente este algoritmo construtivo a outros cenários arbitrários de navegação autônoma, será descrita no capítulo 4 a proposta de um método evolutivo para geração automática das conexões de redes imunológicas artificiais, ou seja, um processo evolutivo de construção e coordenação de estruturas modulares que trabalham com processamento distribuído.

Capítulo 3

Sistemas Imunológicos Artificiais e Algoritmos Evolutivos

Neste capítulo, são evidenciados os conceitos fundamentais da área de imunologia que possibilitaram a aplicação dos Sistemas Imunológicos Artificiais e dos Algoritmos Evolutivos na concepção do sistema híbrido a ser proposto no Capítulo 4.

Inicialmente, na Seção 3.1, são apresentadas as óticas pela quais o sistema imunológico pode ser observado. Na Seção 3.2, é apresentada uma breve descrição dos fundamentos de imunologia importantes para o entendimento do trabalho, como os conceitos de anticorpos e antígenos, e os mecanismos que os envolvem.

A partir das informações da Seção 3.2, na Seção 3.3 é descrita a Teoria da Rede Imunológica e o modelo de rede utilizado nesta dissertação. Este modelo será empregado na formalização do mecanismo de coordenação sob a visão do paradigma de Robótica Baseada em Comportamento (RBC).

Na Seção 3.4, será descrita a forma de representação e implementação computacional dos SIAs, que possibilitará a fixação dos conceitos e a aplicação de redes imunológicas em problemas computacionais.

Na Seção 3.5, são apresentados os algoritmos evolutivos, evidenciando sua característica populacional, codificação e os operadores de recombinação (crossover) e de variação (mutação). Também serão apresentados os conceitos de pressão seletiva e de manutenção de diversidade, que podem ser diretamente associados à capacidade de exploração e exploração do espaço de busca. Ainda nesta seção, é apresentado o algoritmo evolutivo básico.

3.1 Auto-reconhecimento e auto-afirmação

Juntamente com outros sistemas presentes em organismos superiores, o sistema imunológico possui um papel importante na manutenção da integridade do organismo.

Existem duas principais teorias para interpretar o papel do sistema imunológico: auto-reconhecimento e auto-afirmação (BERSINI, 2002).

No caso da teoria do auto-reconhecimento (TAUBER, 1994), defende-se a perspectiva de que o sistema imunológico é responsável por fornecer ao organismo a noção do que ou quais são os seus elementos próprios, ou seja, a detecção do próprio. A partir da capacidade de auto-reconhecimento, é possível implementar, então, a detecção do não-próprio, por processos de discriminação. Dentro do conjunto de elementos não-próprios estão incluídos agentes externos ao organismo, como vírus, bactérias, fungos e parasitas, ou ainda células do próprio organismo com anormalidades estruturais e funcionais, as quais devem ser detectadas e eliminadas antes que causem danos irreversíveis ao organismo. Nesse contexto, todo elemento não-próprio vai causar uma resposta imunológica, mesmo que não apresente um comportamento de um patógeno, ou seja, de uma agente causador de doença.

Por outro lado, a teoria da auto-afirmação (BERSINI, 2002) define o sistema imunológico como um sistema capaz de realizar duas ações antagônicas: tolerar e reagir. Todos os elementos que entram em contato com o sistema imunológico podem ser tolerados, no sentido de não promover uma resposta imunológica (tolerância), ou então podem promover uma resposta imunológica (reação). O que leva à tolerância ou à reação é completamente dependente de todas as experiências de tolerância e reação já vividas pelo sistema imunológico ao longo de sua existência. Sendo assim, não se justifica a caracterização do próprio e do não-próprio, mas sim a existência de tolerância e reatividade. É evidente que o que corresponde ao próprio no caso da teoria do auto-reconhecimento deve ser tolerado, embora esta tolerância não ocorra pelo fato do 'próprio ser o próprio'. De forma equivalente, um patógeno não leva a uma resposta imunológica por ser não-próprio, mas pela ausência de tolerância. É também evidente que se o sistema imunológico tolerasse patógenos, a manutenção da vida do organismo ficaria comprometida.

Ao mesmo tempo em que a teoria da auto-afirmação é mais plausível, por não requerer que o sistema imunológico desenvolva explicitamente mecanismos de auto-reconhecimento, a teoria do auto-reconhecimento é mais pragmática, por permitir associar diretamente o comportamento do sistema imunológico a funcionalidades básicas.

Em toda aplicação em que as funcionalidades dos módulos constituintes do sistema imunológico já estão definidas a priori, a teoria do auto-reconhecimento se apresenta como a mais adequada para se produzir modelos artificiais de sistemas imunológicos capazes de atingir diretamente as funcionalidades pretendidas.

No contexto desta dissertação, a teoria da auto-afirmação se mostra mais apropriada. Embora os anticorpos estejam definidos a priori, o comportamento da rede imunológica vai depender do histórico de atuação ao longo do tempo, sendo que o papel a ser atribuído a cada anticorpo não está definido a priori. Além disso, a rede imunológica pode apresentar atividades regulatórias independente da existência de estímulos externos, pela simples imposição de uma variação no nível de concentração de um anticorpo da rede.

Esses processos computacionais inspirados em conceitos e teorias imunológicas estão vinculados à área de pesquisa denominada Sistemas Imunológicos Artificiais, caracterizada primordialmente pelo desenvolvimento da computação inspirada na imunologia. Os algoritmos associados a esta nova área de pesquisa geralmente exploram mecanismos e conceitos específicos, como o princípio da seleção clonal, a maturação de afinidade e a teoria da rede imunológica (DE CASTRO & TIMMIS, 2002).

3.2 Fundamentos de Imunologia

Tomando como referência o caso particular dos mamíferos, seu sistema imunológico é complexo e apresenta um nível elevado de organização. Trata-se de um sistema distribuído, envolvendo vários órgãos, e que opera por intermédio de um grande número de moléculas e células. Existem mecanismos reguladores altamente sofisticados e o desempenho de todo o sistema imunológico está vinculado ao resultado de variados processos de competição e cooperação entre os agentes constituintes (JANEWAY *et al.*, 1999).

Em face da necessidade de analisar e compreender o sistema imunológico, inúmeras teorias já foram propostas visando descrever os mecanismos complexos responsáveis por tamanha organização e eficiência na execução de diversas tarefas, como reconhecimento de padrões, cognição, memorização e manutenção de diversidade.

O conceito de antígeno é fundamental para compreender a operação do sistema imunológico, particularmente quando se emprega a teoria do auto-reconhecimento. Por antígeno entende-se todo elemento capaz de gerar uma reação imunológica, a qual provoca uma resposta adaptativa do sistema. Como exemplos de antígenos, pode-se mencionar células do próprio organismo que apresentam algum tipo de alteração estrutural ou funcional e agentes externos, como vírus e bactérias. Também sob a perspectiva do auto-reconhecimento, a noção do que é próprio e do que é não-próprio do organismo é indispensável para o bom funcionamento do sistema imunológico, evitando a ausência de reação junto a agentes nocivos e também impedindo que se combata células do próprio organismo, o que caracterizaria doenças auto-imunes.

A teoria da auto-afirmação, por sua vez, não contesta a existência de reação a agentes nocivos e a tolerância às células do próprio organismo. Apenas estes comportamentos não são motivados pela procedência do agente, ou então por propriedades intrínsecas dos mesmos, e sim pela coerência do próprio sistema imunológico com o seu histórico de comportamento, ou seja, sua auto-afirmação.

O sistema imunológico pode ser decomposto em duas partes: o sistema imune inato e o sistema imune adaptativo. O sistema inato representa a primeira linha de defesa do organismo, sendo composto por células fagocitárias e alguns tipos de enzimas. As barreiras físicas do organismo, como a pele nos seres humanos, podem também ser consideradas constituintes do sistema inato. Além de ser responsável pelas primeiras fases de combate às infecções, o sistema imune inato também exerce um papel importante na ativação e regulação do sistema imune adaptativo.

A maior parte das propriedades mais elaboradas sob o ponto de vista de processamento de informação são realizadas pelo sistema adaptativo, o qual fornece ao organismo a capacidade de aumentar a eficácia de suas estratégias de defesa imunológica, adaptando suas células a responderem de forma mais eficiente quando um determinado antígeno é apresentado. Isto acontece porque o sistema altera a estrutura e a concentração de suas células através de um processo conhecido como Princípio da Seleção ou Expansão Clonal (BURNET, 1959).

Este princípio é a mais aceita teoria seletivista para explicar como o sistema imune adaptativo é capaz de produzir novos anticorpos e aumentar sua concentração (relativamente aos demais) em resposta a virtualmente qualquer antígeno, natural ou sintético. Este também é o caminho para se explicar o motivo pelo qual o sistema imunológico é muito eficaz no combate a reinfecções por um mesmo antígeno ou por antígenos similares a outros já apresentados ao sistema imunológico, conhecido como memória imunológica. Este processo é observado através das vacinas, hoje amplamente utilizadas pelos órgãos de saúde, que apresentam ao sistema imunológico os agentes causadores de doenças, devidamente manipulados para perderem suas propriedades patológicas, fazendo com que a memória imunológica seja criada, ampliando o poder de ação no caso de um futuro contágio.

As células mais importantes do sistema imune adaptativo são os linfócitos-B e os linfócitos-T. Ambos são produzidos na medula óssea, mas os linfócitos-T migram para o timo, onde sofrem um processamento antes de estarem prontos para atuar como células imunológicas. Os linfócitos-B são responsáveis pela produção de anticorpos, que são mediadores de um dos principais tipos de resposta imune: a resposta humoral. Anticorpo é o nome dado à molécula de superfície do linfócito-B quando liberada em solução. Portanto, o receptor do linfócito-B pode ser encontrado anexo à superfície da célula ou livre em solução sob a forma de anticorpo. Os anticorpos possuem o importante papel de reconhecer e se ligar aos antígenos, como ilustrado na Figura 3.1. Já os linfócitos-T são mediadores de respostas de base celular.

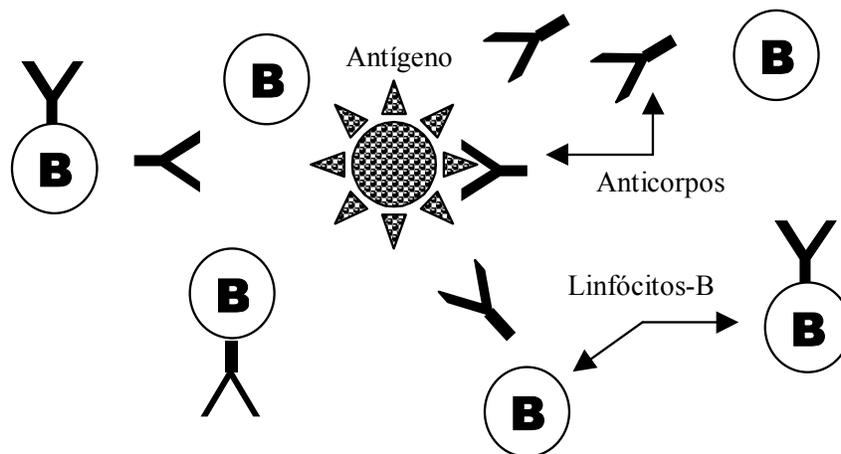


Figura 3.1 – O processo de reconhecimento antigênico pelos linfócitos-B.

Procurando enfatizar apenas as respostas mediadas pelos linfócitos-B, através do aumento da concentração de anticorpos, o sistema imunológico é capaz de disparar vários processos em paralelo, responsáveis pela atenuação e posterior eliminação da ação do antígeno.

Todo linfócito-B possui em sua superfície moléculas de anticorpo. Estas moléculas são capazes de sinalizar para outros componentes do sistema imunológico, por exemplo, células fagocitárias, a presença de antígenos. No entanto, apenas anticorpos que apresentam afinidade físico-química com alguma parte da superfície do antígeno é que podem realizar tal sinalização, sendo que o grau de afinidade físico-química a algum antígeno é que deve guiar o processo de expansão clonal de linfócitos-B. Durante esse processo de clonagem, os novos linfócitos-B podem apresentar modificações em seus anticorpos que tendem a aumentar o grau de afinidade ao antígeno.

Favorecendo continuamente a clonagem dos linfócitos-B cujos anticorpos apresentam maior afinidade ao antígeno, haverá sempre uma tendência de incremento na qualidade da resposta imune. Este processo adaptativo de maturação de afinidade promove a capacidade do sistema imunológico em responder a praticamente qualquer tipo de antígeno, embora requeira um intervalo de tempo significativo até que se alcance um nível de sinalização eficaz.

Um fato importante em relação aos antígenos é que eles apresentam em sua superfície várias estruturas com diferentes características físico-químicas, que compõem a parte reconhecida pelos anticorpos, chamadas de assinaturas. Essa característica aumenta o poder do sistema imunológico, dado que reconhecendo apenas uma ou poucas delas é possível dar início ao processo de resposta imunológica. Além disso, diferentes anticorpos podem reconhecer o mesmo antígeno.

A Figura 3.2 apresenta uma visão simplificada do comportamento global do sistema imunológico. Inicialmente, os antígenos são processados por uma célula apresentadora de antígeno (e.g. macrófago) que possui em sua superfície um complexo CHP/peptídeo, onde CHP é uma sigla para Complexo de Histocompatibilidade Principal, que irá estimular os linfócitos-T a atuarem. Quando um linfócito-T possui alguma afinidade físico-química com peptídeos antigênicos apresentados pelo complexo CHP/peptídeo, ele é estimulado e sinaliza outras células do sistema imunológico a atuarem, em particular os linfócitos-B. Estes últimos irão secretar anticorpos específicos ao antígeno reconhecido.

Após essa breve descrição dos principais componentes e mecanismos envolvidos numa resposta imune, já existem condições mínimas para iniciar a descrição da teoria da rede imunológica. Para uma apresentação mais completa e detalhada dos fundamentos básicos de imunologia, inclusive contando com referências frequentes à literatura mais específica, pode-se recorrer ao texto de DE CASTRO & TIMMIS (2002).

Peptídeos antigênicos que, ligados a moléculas do Complexo de Histocompatibilidade Principal (CHP), serão apresentados na superfície celular

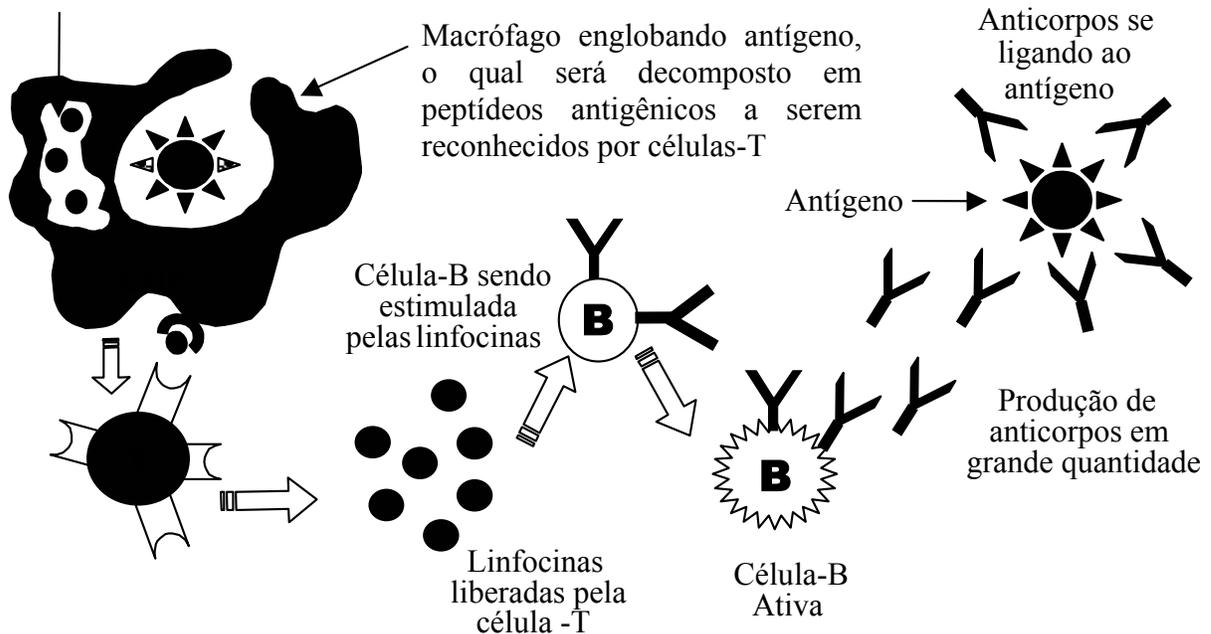


Figura 3.2 – Esquema simplificado da atuação do sistema imunológico. Adaptado de DE CASTRO (2001).

3.3 Teoria da Rede Imunológica

3.3.1 A Inspiração

Muitas foram as teorias propostas para explicar a operação do sistema imunológico. Um dos pontos principais era identificar como ocorria a adaptação ou memorização de um dado agente patogênico, uma vez apresentado ao organismo. Dentre as teorias seletivistas de formação de anticorpos, Niels K. Jerne elaborou a hipótese de que, mesmo sem a exposição a qualquer antígeno, o sistema imunológico era capaz de criar anticorpos com novas estruturas moleculares e que a seleção dos anticorpos dependia diretamente do grau de afinidade físico-química (complementaridade de propriedades físico-químicas) entre o anticorpo e o antígeno, fazendo com que a qualidade da resposta imunológica dependesse da concentração deste anticorpo no organismo.

Desse modo, concluiu-se que uma exposição prévia a um antígeno poderia melhorar o desempenho do sistema quando se desse uma segunda exposição. Além disso, BURNET (1959) levantou a hipótese de que cada anticorpo possui um tipo muito específico de estrutura molecular, o que faz com que a seleção entre vários tipos de anticorpos ocorra diretamente em função do estímulo do antígeno. Uma vez selecionado o anticorpo, os

linfócitos-B que o contiverem irão se proliferar (princípio da seleção ou expansão clonal, já descrito anteriormente) e produzir anticorpos em grande quantidade. Além disso, Burnet introduziu a idéia de clonagem de linfócitos-B, sujeita a processos de hipermutação e edição de receptores, e também a de que toda célula-B que contém anticorpos capazes de reconhecer células do próprio organismo (células auto-reativas) deve sofrer supressão. Com a comprovação experimental dessas hipóteses, JERNE (1974) propôs uma nova teoria para explicar como o sistema imunológico desenvolve tolerância a estruturas próprias do organismo. De posse dessas informações, Jerne observou que certas moléculas que compunham os anticorpos, denominadas idiotopos, eram reconhecidas por outros anticorpos, e elaborou a teoria da rede imunológica ou da rede idiotípica (JERNE, 1984).

3.3.2 A Ferramenta

Tendo em vista a grande quantidade de mecanismos e processos envolvidos no sistema imunológico, cientistas e engenheiros abstraíram ferramentas que levam em consideração características altamente desejadas em sistemas de engenharia e em ambientes computacionais, como: tolerância a falhas, distribuição da informação, gerenciamento descentralizado, reconhecimento de padrões e adaptação (DE CASTRO & VON ZUBEN, 2001; DE CASTRO, 2001; FORREST, 1997).

A teoria da rede imunológica é aquela, dentre as formalizações matemáticas de sistemas imunológicos artificiais, que mais se aproxima da expressão conjunta de todas essas características, já tendo sido aplicada nos mais variados contextos. As redes imunológicas foram aplicadas com resultados expressivos junto a problemas de mineração de dados (DE CASTRO & VON ZUBEN, 2001; NASRAOUI *et al.*, 2002), comunicação entre servidores Web (SUZUKI & YAMAMOTO, 2000), diagnóstico (CAYZER & AICKELIN, 2002), controle para navegação autônoma de robôs (ISHIGURO *et al.*, 1996; MITSUMOTO *et al.*, 1995, 1996, 1997; LEE *et al.* 1997, 1999; WATANABE *et al.*, 1998, 1999), servomecanismo (ISHIGURO *et al.*, 1994).

Para melhor compreender o funcionamento das redes imunológicas, deve-se primeiro ganhar familiaridade com características das células que compõem o sistema imunológico, mais especificamente dos mamíferos, e aos processos desenvolvidos por estas células. Esse sistema possui componentes com diferentes características, dentre os quais destacam-se os macrófagos, os linfócitos-B e linfócitos-T, ou simplesmente células-B e células-T.

Os macrófagos (veja Figura 3.3) são um tipo de fagócito (célula que realiza fagocitose, ou seja, que ingere outras células e moléculas) que compõem o sistema imune inato. Já as células-B podem variar de quantidade (aumentar seu nível de concentração) e modificar as estruturas de suas moléculas receptoras, anticorpos, para promoverem o reconhecimento e combate mais eficiente ao antígeno que está sendo apresentado. Sob a perspectiva da teoria da rede imunológica, os anticorpos possuem duas estruturas fundamentais, os paratopos (parte do anticorpo que pode reconhecer antígenos) e os idiotopos (parte do anticorpo que pode ser reconhecida por outros anticorpos). A Figura 3.4 ilustra a presença de idiotopos e paratopos nas moléculas de anticorpo.

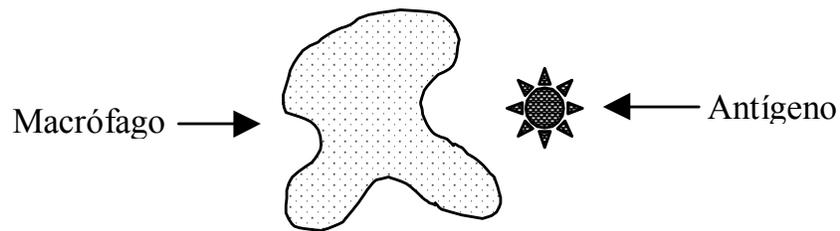


Figura 3.3 – Observação do início do processo de fagocitose. Os macrófagos englobam o agente que é destruído através de processos físico-químicos.

As células-T podem aumentar a sua concentração caso haja o reconhecimento de um antígeno. Elas participam da primeira parte do processo, quando há o reconhecimento do antígeno (veja Figura 3.2). Após essa etapa, seguem-se o aumento da concentração deste tipo de célula e a liberação de uma substância denominada linfocina. A linfocina serve como sinalizador para que células-B com afinidade ao antígeno sejam ativadas, sofram o processo de seleção clonal e produzam plasmócitos, os quais são responsáveis pela liberação dos anticorpos em grande quantidade. Algumas das células-B e células-T envolvidas no processo são mantidas após o término da ação do sistema imunológico e transformam-se em células de memória, a serem requisitadas caso ocorram infecções posteriores pelo mesmo antígeno, ou um antígeno com características similares.

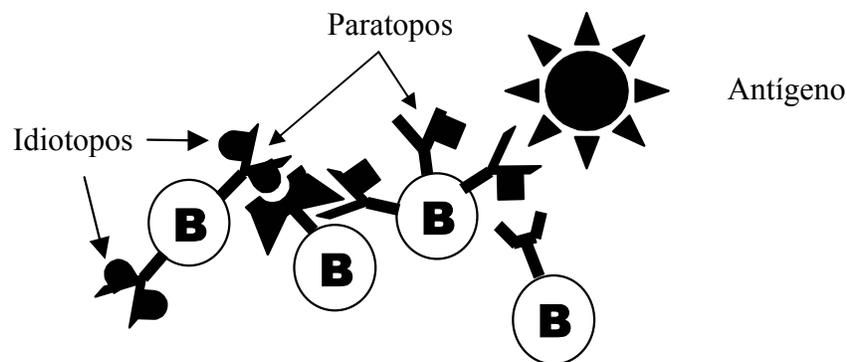


Figura 3.4 – Exemplo pictórico de como se formam as redes imunológicas. Nota-se as conexões entre os paratopos de algumas células-B e os idiotopos de outras, assim como o reconhecimento do antígeno.

A Figura 3.5 ilustra anticorpos de células-B reconhecendo idiotopos de outras células. Esse processo se dá pela afinidade físico-química entre as estruturas.

As redes imunológicas possuem um comportamento que pode ser expresso através das seguintes equações derivadas do modelo proposto por FARMER *et al.* (1986):

$$\frac{da_i(t)}{dt} = \left(\alpha \sum_{j=1}^N m_{ji} a_j(t) - \beta \sum_{k=1}^N m_{ik} a_k(t) + \lambda m_i - k_i \right) a_i(t), \quad i = 1, \dots, N. \quad (3.1)$$

A Equação (3.1) determina a variação no tempo t do nível de concentração do anticorpo i , $a_i(t)$, e seus termos são descritos como segue:

- N é o número de anticorpos que compõem a rede imunológica;
- m_i é a afinidade entre o anticorpo i e um dado antígeno;
- m_{ji} representa a afinidade entre os anticorpos j e i , ou seja, o nível de estimulação;
- m_{ik} representa a afinidade entre os anticorpos i e k , ou seja, o nível de supressão;
- k_i é o coeficiente de morte natural do anticorpo i .

Os coeficientes α , β e λ são responsáveis por indicar com que intensidade os termos atuam. A Equação (3.1) será tratada com maiores detalhes no Capítulo 4, onde será descrito o funcionamento do sistema proposto. A Equação (3.2) é utilizada em vários sistemas que trabalham com redes imunológicas (ISHIGURO *et al.*, 1996; LEE *et al.* 1997, 1999; WATANABE *et al.*, 1999), para normalizar os valores das concentrações, que neste trabalho têm seu valor máximo em 1.

$$a_i(t) = \frac{1}{1 + \exp(0,5 - a_i(t))} \quad (3.2)$$

Note que a Equação (3.1) leva em consideração a importância das conexões entre os nós da rede imunológica (anticorpos e/ou antígenos), de modo a definir o nível de concentração de um dado tipo de anticorpo de acordo com a teoria proposta por JERNE (1974).

Para viabilizar sua implementação computacional, o sistema de equações diferenciais acopladas, todas representadas na forma da Equação (3.1), deve ser aproximado por um sistema de equações a diferenças acopladas, empregando técnicas de cálculo numérico.

3.4 Aspectos de Representação e Implementação Computacional

Por um Sistema Imunológico Artificial (SIA) entende-se a abstração dos princípios, processos e teorias referentes ao sistema imunológico biológico, visando sua reprodução em computador e aplicação como ferramenta de solução de problemas. Essa abstração pode ser decomposta em três partes (DE CASTRO & TIMMIS, 2002):

1. representação esquemática dos componentes do sistema;
2. definição do conjunto de mecanismos utilizados para avaliar a interação dos indivíduos, e entre os indivíduos e o ambiente;
3. emprego de procedimentos de adaptação.

Decisões distintas junto a cada uma das três partes constituintes do processo de abstração conduzem a diferentes propostas de sistemas imunológicos artificiais. Relacionada com a representação, existe a proposta de PERELSON & OSTER (1979), os quais empregaram representações utilizando cadeias de atributos para modelar células e moléculas do sistema imunológico. A idéia principal é utilizar esta representação no desenvolvimento de métricas capazes de avaliar o nível de afinidade (grau de interação) entre os anticorpos e entre anticorpos e antígenos.

Para tanto, a segunda parte do processo de abstração procura relacionar os estímulos do ambiente com funções de avaliação de afinidade dos anticorpos. Assim, é possível se determinar, de forma relativa, a qualidade dos indivíduos (anticorpos) que compõem a população. Utilizando a representação de cadeias binárias para as moléculas receptoras, pode-se aplicar técnicas alternativas na comparação entre as cadeias e, assim, quantificar o nível de afinidade ou reconhecimento entre elas.

Dispondo de uma representação e tendo sido definido um processo de interação dos componentes do sistema, implementa-se a etapa de adaptação, de acordo com os propósitos da aplicação. Na fase de adaptação, estão incluídos todos os procedimentos que capacitam o sistema a executar mudanças quantitativas e qualitativas nos componentes e nas interações dos componentes do sistema imunológico artificial.

A Seleção Clonal (BURNET, 1959), a Seleção Negativa (KRUISBEEK, 1995), a Rede Imunológica (JERNE, 1974; JERNE, 1984), dentre outros, são exemplos de princípios e teorias que vêm sendo amplamente utilizados na produção de sistemas imunológicos artificiais.

O modelo de rede descrito na seção anterior é um exemplo de um sistema imunológico artificial que pode empregar a representação em cadeias binárias para anticorpos e suas moléculas, de modo que a afinidade entre elas possa ser medida pela distância de Hamming, métrica normalmente adotada quando se opera com a representação binária. Seu procedimento de adaptação pode então ser provido por um sistema de equações diferenciais acopladas, onde cada equação componente assume a forma da Equação (3.1). Deste sistema, extrai-se a variação no tempo do nível de concentração de cada anticorpo que compõe a rede. O papel da rede imunológica em uma determinada aplicação certamente estará vinculado ao comportamento temporal do nível de concentração dos anticorpos.

3.5 Algoritmos evolutivos

Há muito tempo sabe-se da capacidade dos mecanismos da natureza na manutenção do equilíbrio de ecossistemas. Dentre esses mecanismos, Darwin observou com atenção o que chamou de seleção natural. Essa seleção explora o diferencial de adaptação ao meio dos indivíduos que constituem uma dada população e propicia, aos mais adaptados, maior capacidade de continuação na espécie, através de processos reprodutivos. Todo esse processo tem como consequência a evolução dos indivíduos da espécie, a fim de torná-los ainda mais adaptados a ambientes invariantes no tempo, ou então a fim de permitir adaptação continuada a ambientes variantes no tempo.

Com base nesses conceitos de seleção e evolução, em ambiente natural, foram concebidos os algoritmos evolutivos, em ambiente computacional. Esses algoritmos partem de uma definição de um espaço de busca, em que se encontram soluções candidatas de um problema a ser resolvido, e simulam processos evolutivos visando detectar regiões promissoras no espaço de busca, em que se encontram soluções candidatas mais capazes de resolver o problema. Se o problema for interpretado como o ambiente e a capacidade de resolução for associada à adaptabilidade ao ambiente, então fica estabelecida uma analogia entre o mundo natural e o mundo virtual em computador.

A seguir, são destacadas algumas propriedades básicas que caracterizam os algoritmos evolutivos (BÄCK *et al.*, 2000a).

Cada indivíduo é passível de descrição na forma de uma lista de atributos. Esses indivíduos representam pontos no espaço de busca, ou espaço de atributos, constituindo possíveis soluções para o problema em questão. A representação de cada indivíduo na forma de uma lista de atributos também se deve a uma analogia direta, desta vez com o cromossomo biológico. Esses cromossomos na forma de listas de atributos podem ser interpretados como um arranjo de genes, sendo que cada gene corresponde a um atributo. O resultado da interpretação de todos os genes é o fenótipo do indivíduo. Em ambientes computacionais, é desejável que exista um único mapeamento entre o genótipo e um fenótipo correspondente.

Os algoritmos evolutivos atuam sobre uma população de indivíduos e empregam operadores genéticos junto ao processo de reprodução, dando maior prioridade aos indivíduos com maior grau de adaptação. Em seguida, é feita a seleção de indivíduos que irão compor a próxima geração. Os operadores genéticos geralmente correspondem a analogias diretas com mutação e recombinação. A mutação corresponde à cópia com certo nível de erro de um indivíduo para a próxima geração. A recombinação implementa a troca de valores de atributos entre dois ou mais indivíduos da população.

Para guiar o processo de busca, é utilizado apenas o valor de adaptação, também denominado *fitness*, dos indivíduos pertencentes à geração atual. Este valor vai estar, de alguma forma, vinculado à capacidade de resolução do problema que cada indivíduo (ou solução candidata) apresenta. Baseado no *fitness* de cada indivíduo, o operador de seleção favorece os indivíduos mais adaptados a propagarem seus atributos à próxima geração.

3.5.1 Tipos de algoritmos evolutivos

Com base nas escolhas efetuadas na especificação da representação dos indivíduos (definindo o espaço de busca), dos operadores genéticos e da forma como a população é manipulada, os algoritmos evolutivos podem ser divididos nos seguintes tipos: algoritmos genéticos, estratégias evolutivas, programação evolutiva e programação genética.

Os algoritmos genéticos (HOLLAND, 1975, 1992) preocupam-se predominantemente com a recombinação, utilizando o operador de crossover a taxas elevadas e aplicando a mutação a taxas reduzidas. O processo de seleção é normalmente realizado de maneira probabilística, proporcional ao *fitness* dos indivíduos. Nas versões mais clássicas dos algoritmos genéticos, os atributos são codificados na forma binária. Este será o tipo de algoritmo evolutivo a ser empregado nesta dissertação.

Nas estratégias evolutivas (RECHENBERG, 1973, 1994), geralmente aplica-se mutação com distribuição normal para a modificação de vetores com valores reais. O operador de mutação é visto como essencial para que o espaço de busca seja varrido de maneira satisfatória. O operador de seleção é determinístico e o tamanho da população, assim como o número de descendentes a cada geração, é fixo.

A programação evolutiva (FOGEL *et al.*, 1966) adota como operador apenas a mutação, descartando a recombinação dos indivíduos. Assim como nas estratégias evolutivas, a mutação é de distribuição normal e o processo evolutivo é estendido aos parâmetros da estratégia. O operador de seleção, nesse caso, é normalmente probabilístico. A programação evolutiva foi inicialmente desenvolvida para trabalhar com máquinas de estado finito, mas atualmente tem várias outras aplicações.

A programação genética (KOZA, 1989, 1992, 1994), por sua vez, pode ser interpretada como uma extensão dos algoritmos genéticos ao tratamento de árvores de atributos, em lugar de listas de atributos. Os operadores genéticos serão específicos e o cromossomo vai apresentar inerentemente tamanho variável. Como os indivíduos da população normalmente correspondem a programas computacionais, o *fitness* de cada indivíduo será definido apenas após a execução do programa, empregando um elenco de instâncias para os argumentos de entrada.

3.5.2 Pressão seletiva e manutenção de diversidade

Dois conceitos muito importantes em algoritmos evolutivos são a pressão seletiva e a manutenção de diversidade. A pressão seletiva é impulsionada pelo operador de seleção. Essa pressão faz com que os códigos dos indivíduos sejam a cada geração mais parecidos, uma vez que a seleção favorece a reprodução dos indivíduos mais adaptados. A manutenção de diversidade vem na contramão da pressão seletiva, pois com a pressão seletiva a queda da diversidade é praticamente inevitável. O operador de mutação pode apresentar um efeito significativo na manutenção da diversidade. É importante notar que, quando a diversidade cai a valores muito baixos, o processo evolutivo perde a capacidade de exploração do espaço de busca.

Os conceitos de pressão seletiva e de manutenção de diversidade podem ser diretamente associados à capacidade de exploração e exploração do espaço de busca (ATMAR, 1994). Dado que o propósito de um algoritmo evolutivo é detectar regiões promissoras do espaço de busca, a manutenção de diversidade é requerida no sentido de assegurar um bom nível de exploração do espaço de busca. Por outro lado, uma vez detectada uma região promissora, a pressão seletiva pode concentrar a busca nesta região, o que corresponde a um procedimento de exploração.

3.5.3 Codificação

Os indivíduos da população representam algumas das possíveis soluções para o problema. É por isso que o número de indivíduos ou o tamanho da população é tão importante quanto as taxas dos operadores. Populações com tamanho reduzido tendem a restringir o potencial da busca e assim promovem uma queda na eficácia do processo evolutivo.

Para que seja possível a aplicação dos operadores, é necessário que cada candidato à solução do problema seja codificado em estruturas adequadas, normalmente uma lista de atributos. Assim, a codificação leva em conta as características mais representativas que devem ser expressas pelos candidatos à solução do problema. Essa representação pode empregar atributos binários, mais comum, inteiros ou com valores no eixo dos reais (veja Figura 3.6).

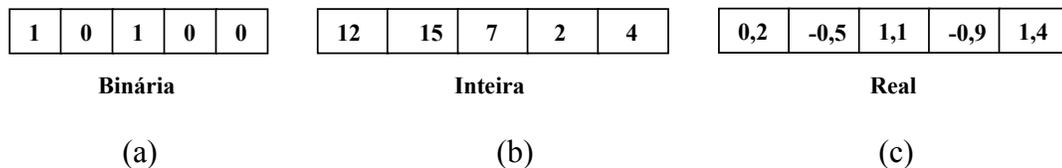


Figura 3.6 – Exemplos de codificação binária, inteira e real.

Como um exemplo, a lista de atributos na Figura 3.6(a) pode representar o conjunto de estados de 5 chaves seccionadoras em uma rede de distribuição de energia elétrica. O bit ‘1’ indicaria chave fechada e o bit ‘0’ indicaria chave aberta. Se o problema a ser resolvido for a minimização das perdas de transmissão de energia elétrica, então vai existir um valor de *fitness* associado a este candidato à solução do problema, diretamente vinculado ao nível de perdas obtido com a configuração de chaves descrita pela lista de bits.

No caso da Figura 3.6(b), a seqüência de valores inteiros pode corresponder a uma proposta de ordenação de elementos de uma lista. Um exemplo clássico para esta codificação é o problema do caixeiro viajante: se os valores inteiros representam índices das cidades a serem visitadas pelo caixeiro, então a seqüência proposta por um dado cromossomo vai corresponder a uma solução candidata. Medindo-se a extensão do percurso indicado pelo cromossomo (a seqüência de cidades é dada pela ordenação dos índices na lista de atributos), pode-se avaliar a qualidade relativa da proposta de solução.

Já para a codificação real da Figura 3.6(c), os 5 valores reais podem corresponder, por exemplo, a valores de parâmetros ajustáveis em um controlador industrial. Se o problema a ser resolvido for a definição dos melhores valores para os 5 parâmetros de modo a maximizar o atendimento de algum objetivo de controle previamente especificado, então o *fitness* do indivíduo vai estar vinculado ao desempenho do controlador projetado com aqueles parâmetros.

3.5.4 Operadores genéticos

Os operadores genéticos são as ferramentas que realmente propiciam a efetividade na busca. Esses operadores são adaptados à codificação dos indivíduos e ao tipo do problema. Como já mencionado, os operadores são geralmente de dois tipos: crossover (recombinação) e mutação (variação).

Crossover

O crossover é responsável pela recombinação dos códigos dos indivíduos da população, escolhidos para propagarem trechos de seus códigos à próxima geração.

Existem vários tipos de operadores de crossover, contudo pode-se destacar o operador básico, chamado de crossover de um ponto ou simples (veja Figura 3.7). Nesse tipo de crossover, a troca de código entre dois (ou mais) indivíduos é feita a partir da escolha aleatória de um ponto no cromossomo, denominado ponto de corte. Uma extensão muito utilizada é o crossover de dois ou mais pontos.

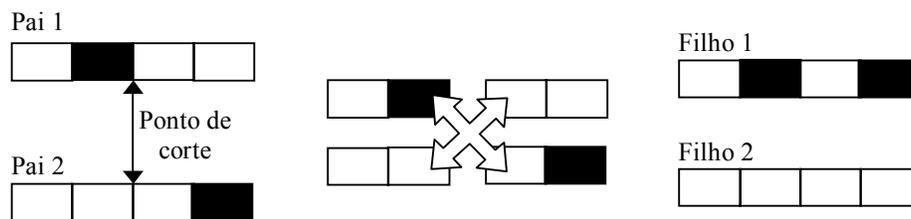


Figura 3.7 – Operador crossover de um ponto ou simples.

Mutação

A mutação é o operador responsável por introduzir variações no código dos indivíduos que farão parte das novas gerações. Essas variações têm por objetivo manter a diversidade dos códigos dos indivíduos que compõem a população. A variação pode ser feita, por exemplo, alterando-se o valor de um dos genes do cromossomo de um indivíduo (veja Figura 3.8). Tal escolha é realizada de maneira aleatória.

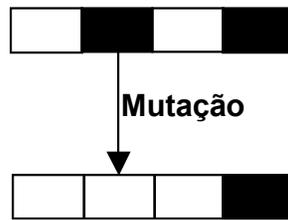


Figura 3.8 – Operador de mutação alterando o valor do gene escolhido.

Nos algoritmos genéticos, a mutação é geralmente aplicada aos indivíduos da nova geração, criados pelo operador de crossover. Existem operadores específicos para representação binária, inteira e real.

3.5.5 Função de Avaliação

Para que a pressão seletiva possa atuar de maneira a levar a população em direção à solução, é necessário saber quais indivíduos ou soluções candidatas são mais adaptados à solução do problema, em uma dada geração do processo evolutivo. Dessa maneira, a função de avaliação ou *fitness* deve ser escolhida de modo a satisfazer o objetivo da evolução, sinalizando na população quais os indivíduos que apresentam maior capacidade de adaptação, ou seja, maior capacidade de resolver o problema.

A definição da função de *fitness* é uma etapa muitas vezes difícil de ser viabilizada, pois em muitos problemas a avaliação depende de fatores nem sempre facilmente identificáveis. Com isso, a escolha da função pode significar o êxito ou não do processo evolutivo.

Este é um dos aspectos mais desafiadores em navegação autônoma de robôs, pois normalmente a função de *fitness* vai estar associada ao desempenho do robô durante a navegação, que pode ser medido em termos do grau de atendimento das tarefas a ele atribuídas. Chegar a um valor de *fitness* confiável pode requerer diversos experimentos de navegação, incrementando custo computacional e tempo de obtenção dos resultados. Esta questão será melhor abordada no Capítulo 4.

3.5.6 O processo de seleção

Existe um processo de seleção para escolha dos indivíduos que irão se reproduzir, sendo que esta seleção é geralmente proporcional ao *fitness* de cada indivíduo. Os novos indivíduos são então submetidos aos operadores de crossover e mutação, ou apenas mutação, e podem passar por um outro processo de seleção, caso os seus progenitores também sejam considerados como candidatos a compor a próxima geração.

O operador de seleção pode ser probabilístico ou determinístico dependendo do objetivo da evolução, das características da população e do problema abordado. Um método de seleção muito utilizado é probabilístico e considera a seleção como sendo diretamente

proporcional aos valores de *fitness* dos indivíduos, sendo denominado de método da roleta (*roulette wheel*) (GOLDBERG, 1989; BÄCK *et al.*, 2000a). Este será o método de seleção empregado neste trabalho.

3.5.7 Algoritmo evolutivo básico

Após a definição da representação dos indivíduos, dos operadores e da função de *fitness*, os algoritmos evolutivos seguem fundamentalmente as mesmas etapas. Descreve-se a seguir um algoritmo evolutivo básico.

```
Algoritmo evolutivo básico;  
Iniciar a população;  
Avaliar os indivíduos da população, atribuindo o valor do fitness;  
Enquanto a condição de parada não for alcançada:  
  {  
    Selecionar os indivíduos a serem reproduzidos com base no fitness;  
    Reproduzir os indivíduos selecionados aplicando recombinação;  
    Efetuar mutação junto aos indivíduos gerados;  
    Inserir os indivíduos gerados na população de acordo com um operador de seleção;  
    Avaliar os indivíduos da população, atribuindo um valor do fitness;  
  }
```

Algoritmo 3.1 – Algoritmo Evolutivo Básico.

A população é normalmente iniciada aleatoriamente, apesar de haver a possibilidade de se escolher outros métodos (SANTOS *et al.*, 2001). A condição de parada deve estar de acordo como o objetivo da evolução, de modo que podem ser escolhidos limiares de *fitness* ou um número de gerações a serem alcançadas.

Caso a condição de parada não seja alcançada, entra em execução o principal laço do algoritmo iniciando o processo de reprodução. Primeiramente, os indivíduos que irão se reproduzir são selecionados, normalmente com base no seu valor de *fitness*. Assim, os indivíduos mais adaptados são favorecidos e tendem a ser escolhidos. Feita a escolha, os indivíduos podem se reproduzir por recombinação, a uma dada taxa. Uma vez criados os indivíduos da nova geração, eles sofrerão a aplicação do operador de mutação a uma dada taxa. Ao final dos processos de reprodução e variação, um outro operador de seleção fará a escolha dos indivíduos que irão compor a próxima geração. Qualquer que seja o critério de

seleção adotado, a escolha é feita com base no valor de *fitness* dos indivíduos. A população é então avaliada e a condição de parada é novamente verificada.

Capítulo 4

Uma Abordagem Imuno-Genética para Navegação Autônoma

Neste capítulo, é apresentado o sistema híbrido proposto. Esse sistema utiliza a dinâmica das redes imunológicas como mecanismo de coordenação dos módulos de comportamento elementar, representados aqui pelos anticorpos.

Antes da apresentação do sistema em si, na Seção 4.1 é feita uma análise das vantagens da utilização de sistemas híbridos.

Na Seção 4.2, é apresentada a proposta de utilização de redes imunológicas no problema de navegação autônoma. Para tanto, são apresentadas as características dos componentes do sistema, assim como sua estrutura e o modelo de rede utilizado. Além disso, a rede imunológica é definida como o mecanismo de coordenação.

Na Seção 4.3, o sistema imuno-genético proposto é ressaltado como um mecanismo de adaptação utilizado na obtenção de padrões de conexões para redes imunológicas que atendam os objetivos da navegação. São descritos os detalhes de representação e da aplicação dos algoritmos evolutivos.

Nas Seções 4.4 e 4.5, são feitas as apresentações do ambiente de simulação utilizado na evolução e teste das redes imunológicas e também do modelo computacional do robô, ambas de maneira introdutória. Maiores detalhes são encontrados no Apêndice A.

4.1 Motivação para uma abordagem híbrida

Em face da disponibilidade crescente de recursos computacionais e de técnicas para processamento de informação e solução de problemas, torna-se viável explorar, em um mesmo dispositivo computacional, aspectos complementares de algumas técnicas, produzindo os chamados sistemas híbridos. Quando estes aspectos complementares são explorados coerentemente no atendimento dos requisitos do problema, os sistemas híbridos vão acabar produzindo soluções com desempenho superior à aplicação isolada, ao mesmo problema, de cada técnica que o compõe.

Portanto, um sistema híbrido pode ser definido como aquele que é sintetizado empregando conjuntamente ferramentas computacionais caracterizadas por etapas de concepção e fontes de inspiração distintas. A distinção de origem dessas ferramentas proporciona a expressão de mecanismos alternativos para processar e armazenar informação.

Em navegação autônoma, muitos sistemas híbridos surgiram da percepção de que a aplicação isolada de certas ferramentas computacionais levava à impossibilidade de se efetuar o controle adequado. FUKUDA & SHIBATA (1994) conceberam um sistema de controle hierárquico aplicando lógica nebulosa, redes neurais e algoritmos genéticos. Essas três técnicas foram combinadas de modo a fornecer ao sistema capacidade de adaptação (redes neurais), controle (redes neuro-nebulosas) e aprendizagem (redes neurais e algoritmos genéticos). Em outras pesquisas com navegação autônoma, foram apresentadas propostas bem sucedidas baseadas em redes neurais e lógica nebulosa (FIGUEIREDO, 1996; CRESTANI, 2001) e redes neurais e algoritmos genéticos (MONDADA, 1995; NOLFI, 1997). Vários outros sistemas híbridos podem ser encontrados, tanto para a navegação autônoma de robôs quanto para sistemas de controle aplicados em outras áreas.

Esta dissertação tem como proposta principal um sistema híbrido que utiliza as técnicas de Sistemas Imunológicos Artificiais e Computação Evolutiva para concepção de um sistema de controle autônomo para robôs móveis. Este sistema será responsável por permitir a um robô navegar por um ambiente visando atender a múltiplos objetivos: coletar lixo e conduzi-lo a um depósito, desviar de obstáculos e manter um nível mínimo de energia interna.

Com o emprego da hibridização, criam-se condições para reunir em uma única ferramenta as características necessárias para o atendimento dos objetivos de navegação. Os sistemas imunológicos artificiais, mais especificamente as redes imunológicas artificiais (FARMER *et al.*, 1986), introduzem a dinâmica necessária para a implementação, além de operarem com informação distribuída e grande capacidade de adaptação. Já as técnicas de computação evolutiva, mais especificamente os algoritmos genéticos (BÄCK *et al.*, 2000a; BÄCK *et al.*, 2000b), fornecem a capacidade de busca por um padrão de conexões para a rede imunológica que, por sua vez, conduza à eficiência e flexibilidade no alcance dos múltiplos objetivos.

Nem todos os aspectos da estrutura da rede serão passíveis de ajuste automático, pois os anticorpos da rede serão definidos a priori. Mesmo assim, a possibilidade de atuar em um elenco mais amplo de aspectos de projeto é a motivação principal para a proposição de uma abordagem híbrida do tipo imuno-genética.

Utilizando o princípio da modularização do processo de síntese a partir da proposição de comportamentos elementares, os quais vão representar os anticorpos da rede imunológica, o sistema imuno-genético vai ser capaz de encontrar uma estrutura de rede, ou seja, determinar automaticamente as conexões entre os anticorpos, de modo que a rede seja capaz de realizar a tarefa de navegação. Assim, no lugar de se buscar a sintonia de uma estrutura de controle previamente fornecida, o que se busca aqui é também propor automaticamente a estrutura de coordenação dos comportamentos elementares.

Esta proposta para a construção de sistemas de controle autônomo garante bom desempenho em situações em que múltiplos comportamentos elementares devem ser integrados das mais variadas formas para se definir a seqüência de ações do robô a cada momento. Em outras palavras, como há múltiplos objetivos a serem atendidos, a definição da seqüência de comportamentos elementares que leva à maximização do atendimento dos objetivos não pode resultar simplesmente de um mecanismo estático de mapeamento. A dinâmica de interação dos anticorpos na rede imunológica, os quais representam regras estáticas do tipo <condição>-<ação>, será então responsável pela coordenação dos comportamentos elementares. E o comportamento dinâmico da rede imunológica será, por sua vez, estabelecido pelo padrão de conexões a ser otimizado via algoritmos evolutivos.

4.2 Uma rede imunológica para navegação autônoma

Antes de apresentarmos a proposta dessa dissertação, é necessário que seja descrito o sistema de navegação autônomo que a inspirou. Esse sistema foi desenvolvido por WATANABE *et al.* (1999) e está baseado na teoria da rede imunológica, apresentada no capítulo 3.

Nesse sistema de navegação, o robô é encarado como o organismo e as informações provenientes dos sensores, que retratam a situação corrente, são tomadas como sendo antígenos. Já os anticorpos são interpretados como os nós da rede. A Tabela 4.1 faz a correlação entre o organismo e o robô.

Tabela 4.1 – Correlação entre um organismo biológico e um robô.

Sistema Imunológico	Sistema Imunológico Artificial
Organismo	Robô
Antígeno	Situação corrente captada pelos sensores
Anticorpo	<condição>-<ação>/Nó da rede e nível de estimulação
Paratopo	<condição>-<ação>
Idiotopo	ID e nível de estimulação
Dinâmica da rede	Sistema de equações diferenciais acopladas
Aumento ou redução na quantidade de anticorpos	Variação na concentração de anticorpos
Sensibilidade do anticorpo	Estimulação e/ou Supressão
Seleção do anticorpo	<i>Roulette Wheel</i> sobre as concentrações

Da teoria de sistemas imunológicos, é sabido que a presença de antígenos representa uma perturbação no estado de equilíbrio da rede imunológica, levando a uma reação imunológica que equivale à redefinição do nível de concentração dos anticorpos.

Portanto, como todo antígeno vai levar a uma reação imunológica, ou seja, ao aumento da concentração de alguns anticorpos, em detrimento de outros, então de toda possível situação corrente de navegação sempre será gerada uma ação do robô, associada a um processo de seleção que toma como argumento o nível de concentração alcançado pelos anticorpos. Espera-se que os anticorpos sejam capazes de lidar adequadamente com os antígenos, de modo a proporcionar ao robô uma navegação que permita alcançar os objetivos previamente determinados.

4.2.1 Abordagens similares na literatura

WATANABE *et al.* (1999) exploraram este comportamento regulatório dos valores de concentração de anticorpos para implementar o processo de tomada de decisão do robô. Uma das maiores limitações da proposta de WATANABE *et al.* (1999) está no fato de que as conexões da rede, responsáveis por guiar a reação imunológica, são definidas através de técnicas construtivas de aprendizado por reforço. Assim, a construção das conexões leva em consideração apenas o aspecto local da tomada de decisão, não havendo domínio suficiente acerca do impacto global que cada modificação local da rede imunológica pode causar. Quanto mais complexa a tarefa de navegação e quanto maior o número de nós da rede imunológica, menos eficazes tendem a ser as técnicas construtivas.

O algoritmo construtivo proposto por WATANABE *et al.* (1999) utiliza um repertório de anticorpos pré-definidos que serão conectados como resposta a eventos verificados ao longo do processo de navegação. Quando uma situação, representada pelos antígenos, entra no sistema, ela gera uma reação com maior ou menor intensidade em cada um dos anticorpos, dependendo da afinidade entre eles e os antígenos. Dentre os anticorpos com maior concentração, é escolhido via *roulette wheel* o anticorpo que irá atuar designando uma ação ao robô. Diante de regras estabelecidas pelo projetista para a avaliação da ação tomada pelo robô, o resultado será avaliado como positivo ou negativo. Caso positivo, ou seja, caso a consequência da ação seja benéfica à trajetória, o anticorpo selecionado será estimulado pelos outros que também foram excitados pelos antígenos. Caso o desempenho seja considerado negativo, o anticorpo será então suprimido pelos demais.

Em uma proposta mais ambiciosa, WATANABE *et al.* (1998) aplicaram um algoritmo evolutivo tanto para definir os anticorpos quanto para estabelecer as conexões. O espaço de busca associado é bem mais amplo, pois não apenas se devem otimizar os nós da rede imunológica, como também as conexões, simultaneamente. Para viabilizar o processo de busca, os autores optaram por uma abordagem distal na definição dos módulos de comportamento elementar, de modo que a tarefa de navegação pudesse ser realizada com sucesso por uma rede imunológica contendo um número bem reduzido de nós e conexões.

Como a proposta desta dissertação pode ser interpretada como uma abordagem mista, com aspectos proximais e distais, as comparações serão realizadas junto à proposta de WATANABE *et al.* (1999), a qual apresenta redes imunológicas com dimensões equivalentes e um conjunto de anticorpos também equivalente. Fixando-se o conjunto de anticorpos, os quais estão associados a comportamentos proximais elementares, o algoritmo evolutivo irá se ocupar apenas com a definição das conexões da rede.

Recorrendo a uma abordagem proximal para a definição dos anticorpos, a rede imunológica vai atuar na tomada de decisão de baixo nível, no estilo <vire à direita> ou < siga em frente>. Por outro lado, com base em uma abordagem distal os anticorpos acabam sendo associados a comportamentos de alto nível do tipo <busque a base de recarga> ou <desvie do obstáculo>. Dois aspectos são evidentes: (i) a quantidade de regras do tipo <condição>-<ação> de baixo nível tende a ser bem maior que a quantidade de regras de alto nível, o que se reflete no número de anticorpos na rede imunológica; (ii) múltiplas decisões seqüenciais em baixo nível precisam ser tomadas para corresponder a uma única decisão de alto nível.

4.2.2 Origem e motivação do problema de navegação

O problema de navegação a ser considerado foi modelado inicialmente por PFEIFER (1995) e consiste na conservação da integridade do robô, associada ao seu nível interno de energia, sem negligenciar o atendimento de outros objetivos. Na concepção de PFEIFER (1995), imagina-se um organismo cibernético enviado a um outro planeta com a missão de coletar cogumelos e, periodicamente, repor seu nível interno de energia para se manter operante.

Assim como esse organismo cibernético enviado em missão, o robô do problema de navegação a ser considerado neste trabalho deve manter seu nível de energia e realizar a tarefa de coleta de lixo no ambiente. Em virtude da existência de múltiplos objetivos, a tarefa de navegação ganha aspectos de complexidade que dificultam sobremaneira a missão do projetista. Sendo assim, se justifica plenamente a proposição de um mecanismo automático de busca do melhor perfil de conexão para a rede imunológica, a qual será responsável pela coordenação de comportamentos elementares de navegação. Além disso, a complexidade da tarefa de navegação vai permitir avaliar devidamente o potencial das redes imunológicas a serem propostas.

4.2.3 Objetivos da navegação

Esta seção busca apresentar uma formalização matemática que permita modelar os objetivos da navegação. O robô deve ser capaz de navegar pelo ambiente coletando lixo, transportando-o até a base e desviando de obstáculos.

A base, além de ser a receptora de lixo, é também o local em que se encontra o módulo capaz de realizar a recarga de energia do robô. Sendo assim, toda vez que o robô deposita lixo ele pode recarregar suas baterias. O nível de energia do robô deve ser priorizado, ou seja, caso a energia seja insuficiente para realizar a tarefa de transporte de lixo, o robô deve retornar à base apenas para realizar a recarga de energia.

A energia interna do robô (E) é atualizada a cada passo e seu decremento envolve 4 fatores: a energia necessária para se movimentar, a energia necessária para capturar e carregar o lixo e uma taxa de energia cobrada toda vez que ocorre colisão contra obstáculos. Assim, a atualização da energia é descrita pela Equação (4.1) para cada iteração t .

$$E(t) = \begin{cases} E(t-1) - E_{passo} - \gamma_1 E_{lixo} - \gamma_2 E_{colisão} - \gamma_3 E_{captura}, & \text{quando não há recarga} \\ E_{max}, & \text{quando há recarga} \end{cases} \quad (4.1)$$

onde $\gamma_1=1$ apenas quando o robô está carregando lixo, sendo igual a zero em outras situações; $\gamma_2=1$ apenas quando o robô sofre uma colisão, sendo igual a zero em outras situações; e $\gamma_3=1$ apenas quando o robô captura uma unidade de lixo, sendo igual a zero em outras situações.

Deve ser observado que a regra de decremento do nível de energia do robô representa algo semelhante ao processo de metabolismo de um organismo vivo. Assim, a decisão da ação a ser tomada não depende apenas da situação corrente, captada através dos sensores externos, mas também da condição interna indicada pelo sensor de energia.

4.2.4 Estrutura do modelo de rede imunológica

O modelo computacional da rede imunológica deve reproduzir o comportamento dinâmico de um sistema imunológico segundo a teoria da rede imunológica. Como descrito no Capítulo 3, há interação mútua entre os anticorpos e entre anticorpos e antígenos.

Quando os antígenos adentram o sistema, a rede imunológica coordena a resposta imunológica através da interação dos anticorpos com o antígeno e de cada anticorpo com os demais anticorpos da rede. O padrão de conexões da rede imunológica é fundamental na determinação do tipo de interação que pode ocorrer. Este processo de interação será modelado na forma de um sistema de equações diferenciais acopladas, tendo como variáveis manipuladas as concentrações dos anticorpos ao longo do tempo. O anticorpo que resultar com a maior concentração terá maior probabilidade de ser selecionado para fornecer a ação de navegação. Isso porque um método probabilístico é utilizado para tal seleção.

Os anticorpos são representações de regras compostas de pares de <condição>-<ação>, conforme apresentado na Figura 4.1. Estas regras correspondem ao paratopo do anticorpo. A outra parte da estrutura, denominada de idiotopo, corresponde às conexões realizadas pelo anticorpo.

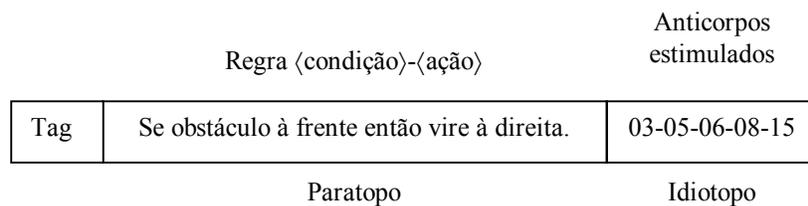


Figura 4.1 – A estrutura do anticorpo é composta pela regra <condição>-<ação> que corresponde à parte denominada paratopo e por suas conexões ou idiotopos.

O padrão de conexões vai ser responsável pela estimulação ou supressão dos anticorpos da rede imunológica, interferindo diretamente na determinação do nível de concentração de cada anticorpo.

4.2.5 Antígenos e Anticorpos

Os antígenos, que correspondem às informações provenientes dos sensores, são codificados de maneira que possam ser comparados com o campo de <condição> dos anticorpos. O número de antígenos é diretamente proporcional ao número de situações captadas do ambiente.

Nesse sistema, os anticorpos são divididos em grupos de comportamento (Seção 2.4.3) que identificam basicamente obstáculos, lixo, base e energia interna. Esses grupos são desmembrados em várias outras condições mais específicas, como demonstra a Tabela 4.2.

Tabela 4.2 – Decomposição em condições específicas

Condição-Antígeno	
Lixo	frente
	esquerda
	direita
	nenhum
	carregando
Obstáculo	frente
	esquerda
	direita
	nenhum
Base	frente
	esquerda
	direita
	nenhuma
Energia interna	perto
	longe
	muito longe
Energia interna	alta
	baixa

Para cada uma dessas quatro possibilidades de <condição>, existe um anticorpo com uma ação associada que irá incidir nos atuadores do robô. As ações são mudanças de direção que determinarão a trajetória segura até os alvos (lixos e bases). A Tabela 4.3 descreve as possibilidades de <ação>.

Tabela 4.3 – Possibilidades de escolha da ação a ser associada a cada condição.

Ação
Vá em frente
Vire à direita
Vire à esquerda
Procure a base
Pegue o lixo
Explore (Opte por uma direção aleatória)

As ações descritas na Tabela 4.3 representam o comportamento do robô quando em navegação. Porém devem ser tomadas algumas ações como padrão. Um exemplo é dado quando o robô alcança a base. Nesse momento, ele tem sua energia recarregada e o lixo, se ele estiver carregando, coletado.

De acordo com essas tabelas, a estrutura dos antígenos deverá ser idêntica à parte que compõe a <condição> nos anticorpos, como mostra a Figura 4.2. De acordo com as situações descritas na Tabela 4.2, têm-se antígenos relacionados com a direção dos obstáculos, direção dos lixos, direção e distância das bases e nível de energia interna.

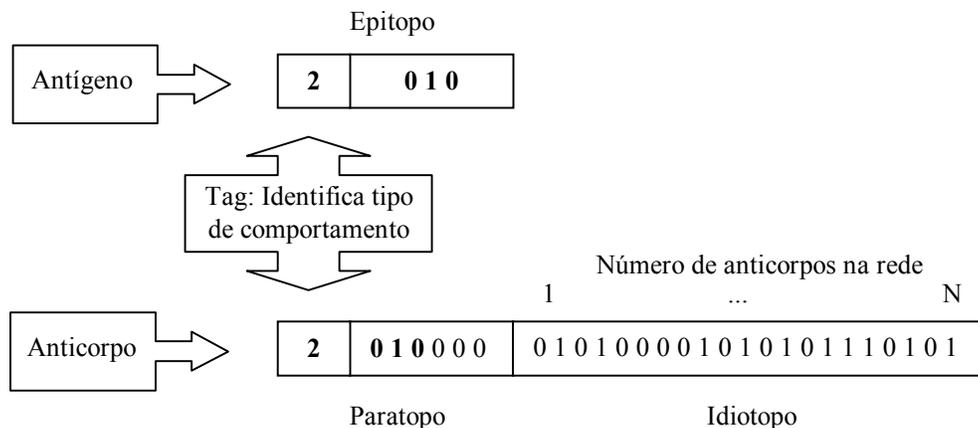


Figura 4.2 – As codificações dos antígenos e dos anticorpos são arranjos binários. O antígeno tem sua estrutura idêntica ao primeiro trecho da estrutura do anticorpo, permitindo a comparação bit a bit para se determinar o grau de afinidade.

Na Figura 4.2, observam-se o antígeno e o anticorpo com a codificação binária empregada. As *tags* ressaltadas na figura são os identificadores, tanto para os antígenos quanto para os anticorpos, sobre o grupo de comportamento ao qual pertencem (lixo, base, obstáculo ou energia).

Os três primeiros bits identificam a posição do objeto: esquerda, frente e direita, respectivamente. Os três bits seguintes, a ação a ser tomada, cujas possibilidades são: ir para frente, direita ou esquerda. Compondo o idiotopo, última parte da codificação, encontram-se as conexões do anticorpo na rede, com o bit 1 na posição i indicando existência de conexão com o i -ésimo anticorpo.

Essas conexões indicam quais são os anticorpos estimulados pelo anticorpo cuja representação está sendo considerada. Portanto, a estimulação pode provir da afinidade, representada por m , com o antígeno e também de outros anticorpos da rede imunológica. No caso desse modelo, o valor de m é 1 para os casos em que há afinidade e 0 caso contrário. A afinidade $m=1$ é dada pela completa identificação do antígeno com o paratopo do anticorpo, caso contrário seu valor é 0. A estimulação proveniente de um anticorpo será diretamente proporcional ao nível de sua concentração.

No sentido contrário ao da estimulação, tem-se a supressão, a qual é exercida pelo anticorpo estimulado. Do mesmo modo que a estimulação, a supressão está relacionada ao nível de concentração do anticorpo.

O nível de concentração do i -ésimo anticorpo em uma iteração t , dado por $a_i(t)$, corresponde, em termos biológicos, à quantidade de um tipo específico de anticorpo que está presente no organismo. No modelo matemático a ser apresentado, o nível de concentração é a variável manipulada diretamente, não derivando, por exemplo, de uma variável que indique o número de anticorpos daquele tipo no organismo.

O modelo aqui utilizado para representação das moléculas de anticorpos, assim como o sistema de equações diferenciais ordinárias que controla a rede imunológica, derivado do modelo proposto por FARMER *et al.* (1986), é apresentado a seguir:

$$\frac{da_i(t)}{dt} = \left(\alpha \sum_{j=1}^N m_{ji} a_j(t) - \beta \sum_{k=1}^N m_{ik} a_k(t) + \lambda m_i - k_i \right) a_i(t), \quad i = 1, \dots, N, \quad (4.2)$$

$$a_i(t) = \frac{1}{1 + \exp(0,5 - a_i(t))} \quad (4.3)$$

onde:

- N é o número de anticorpos que compõem a rede imunológica;
- m_i é a afinidade m entre o anticorpo i e um dado antígeno;
- m_{ji} representa a afinidade m entre os anticorpos j e i , ou seja, o nível de estimulação;

- m_{ik} representa a afinidade m entre os anticorpos i e k , ou seja, o nível de supressão;
- k_i é o coeficiente de morte natural do anticorpo i .

Assim, o termo $\left\{ \alpha \sum_{j=1}^N m_{ji} a_j(t) \right\}$ na Equação (4.2) refere-se ao valor da soma das

estimulações sobre o anticorpo i , dados os anticorpos j sobre os quais ele exerce supressão. O coeficiente α indica com que nível essa estimulação ocorre. Por outro lado, o termo

$\left\{ \beta \sum_{k=1}^N m_{ik} a_k(t) \right\}$ na Equação (4.2) refere-se ao valor da soma das supressões sofridas pelo

anticorpo i , dados os anticorpos k sobre os quais ele exerce estimulação. O coeficiente β indica com que nível essa supressão ocorre.

A estimulação do antígeno sobre o anticorpo i é representada pelo termo λm_i , onde λ é o nível com que a estimulação é exercida. O termo k_i representa o coeficiente de morte natural do anticorpo i . Esse coeficiente faz com que a concentração dos anticorpos menos estimulados fique em níveis inferiores aos demais.

A Equação (4.3) é utilizada para limitar o intervalo de valores admissíveis para as concentrações, que nesse sistema têm seu máximo em 1.

4.2.6 Rede Imunológica: Mecanismo de Coordenação

De acordo com a robótica baseada em comportamento, a rede imunológica realiza aqui o papel de mecanismo de coordenação. O processo tem início quando os antígenos produzem uma perturbação no estado da rede através da resposta imunológica gerada pelos anticorpos. A rede inicia então um processo de auto-organização de forma a convergir novamente a um estado de equilíbrio. Para que isso aconteça, entram em funcionamento as estimulações (promovidas pelos antígenos e anticorpos) e as supressões (promovidas pelos anticorpos). Esse processo finaliza quando o nível de concentração dos anticorpos entra em equilíbrio, em uma nova configuração. De posse desses novos valores de concentração, através do mecanismo de seleção probabilístico, conhecido como *roulette wheel* (BÄCK *et al.*, 2000a), um dos anticorpos é escolhido sendo que o anticorpo mais concentrado terá maior probabilidade de ser selecionado. É importante lembrar que apenas um anticorpo é escolhido em cada iteração t . Portanto, o mecanismo de coordenação é caracterizado por um processo de competição.

Para melhor entendimento, considere uma situação de navegação em que o robô está com um nível baixo de energia, devendo assim priorizar a recarga de suas baterias ao invés de capturar o lixo. A situação é ilustrada na Figura 4.3. Os antígenos formados representam a condição atual de leitura dos sensores.

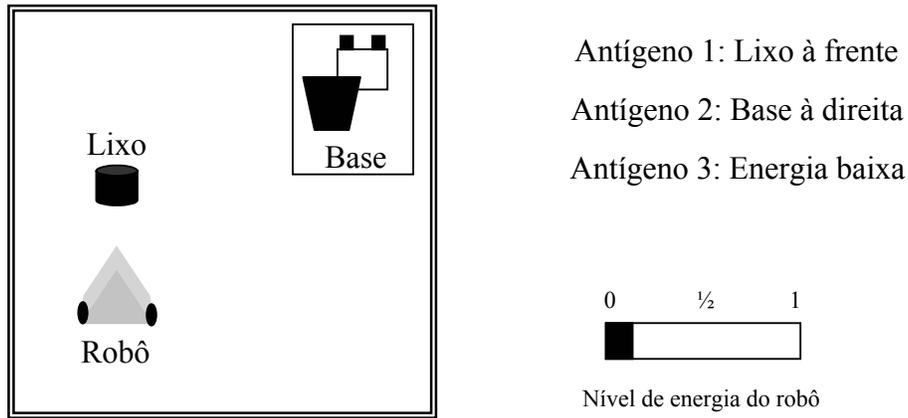


Figura 4.3 – Exemplo de situação em que a tomada de decisão não depende somente das condições externas, mas também do nível de energia. Os antígenos formados representam essa situação.

Se nesse caso o robô fosse controlado por uma rede imunológica, como a ilustrada na Figura 4.4, sua decisão seria a de desviar na direção da base, evitando a captura do lixo, de modo a efetuar a recarga de suas baterias.

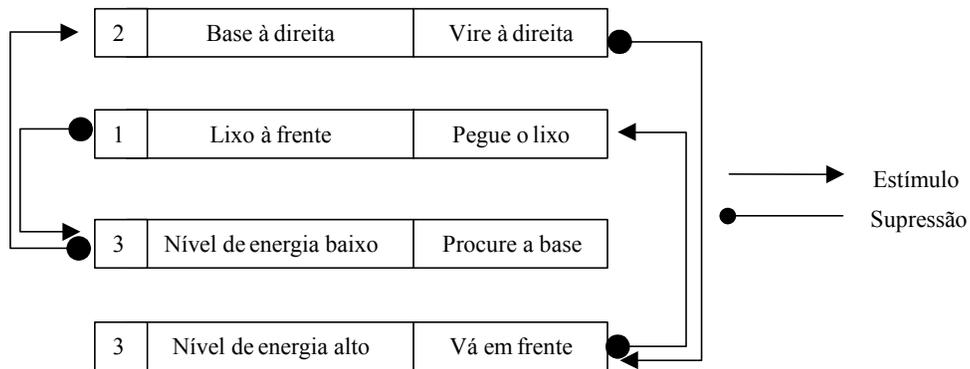


Figura 4.4 – Rede imunológica projetada visando a manutenção da integridade do robô.

Essa dinâmica proporciona ao sistema condições para que haja o aparecimento de comportamentos complexos, uma vez que a rede não responde especificamente a um único antígeno, mas sim a antígenos resultantes do próprio processo de navegação, sempre lembrando que os anticorpos irão apresentar níveis específicos de concentração ao longo do tempo.

Algoritmo

O algoritmo que define o funcionamento do sistema de controle será descrito a fim de explicitar as etapas associadas à ação tomada pelo robô, a cada passo. Em linhas gerais,

o algoritmo tem 3 fases (veja Figura 4.5): aquisição das informações dos sensores, processamento e ação.

A rede imunológica responsável pelo controle está situada na fase de processamento. O processamento compreende a transformação das informações dos sensores em antígenos, a verificação do grau de afinidade entre os anticorpos e os antígenos, o cálculo da concentração com base nessa afinidade, assim como na dinâmica de estímulo e supressão, ditada pelas conexões da rede, e a escolha do anticorpo que irá atuar no robô.

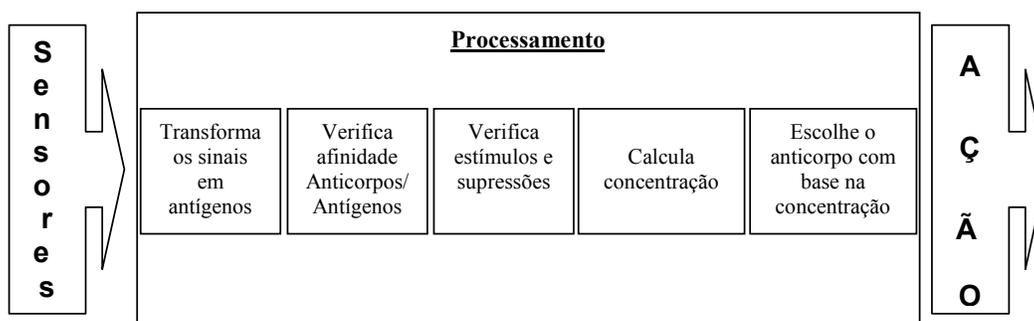


Figura 4.5 – Algoritmo para cada passo do robô, que resulta em uma ação.

4.3 Um mecanismo de adaptação para a rede imunológica

Em seu trabalho, WATANABE *et al.* (1999) salientaram a necessidade de haver um mecanismo de adaptação para a concepção de redes imunológicas artificiais de maneira automática. Isso deve-se à elevada complexidade quando se defronta com o aumento no número de anticorpos. Esse mecanismo normalmente é desmembrado em duas partes: o mecanismo de ajuste e o mecanismo de inovação. Tais mecanismos fazem parte da criação e manutenção dos anticorpos que compõem a rede imunológica.

O mecanismo de ajuste implementado em WATANABE *et al.* (1999) utiliza técnicas construtivas de aprendizado por reforço para ajustar as conexões. Esse mecanismo orienta o sentido da estimulação e, conseqüentemente, da supressão entre os anticorpos excitados em uma dada situação. De acordo com as regras de reforço e penalidade criadas pelo projetista, é recompensado com estímulo o anticorpo que ativar uma ação bem sucedida, e com supressão a ação mal sucedida.

As conexões da rede devem estar adequadamente definidas para que os processos de estimulação e supressão façam surgir comportamentos adequados para cada condição de navegação.

O mecanismo de inovação implementado em WATANABE *et al.* (1999), por sua vez, é baseado em um processo de criação aleatória de novos anticorpos que poderiam integrar a rede. Quando um anticorpo da rede possui uma taxa de concentração abaixo de um

determinado limiar ele é então substituído por outro, que é inserido através do mecanismo de ajuste descrito.

Apesar das potencialidades desta abordagem construtiva, vinculadas à criação de novos repertórios de anticorpos através do mecanismo de inovação, o mecanismo de ajuste utilizado é ineficaz: uma vez que o mecanismo de atualização das concentrações envolve toda a rede imunológica, é difícil isolar o papel de um dado anticorpo, mesmo considerando diversas situações de navegação (NOLFI & FLOREANO, 2000). Em outras palavras, o uso de uma estrutura cognitiva que trabalha com processamento distribuído dificulta sobremaneira a tarefa de construção incremental da estrutura.

Buscando uma alternativa ao mecanismo de ajuste de WATANABE *et al.* (1999), a proposta desta dissertação corresponde a um novo mecanismo de ajuste em que os idiotopos da rede imunológica são definidos através de algoritmos evolutivos, produzindo uma rede capaz de realizar as múltiplas tarefas para as quais foi designada.

A aplicação de computação evolutiva poderia se estender também à criação de um mecanismo de inovação, ou seja, à proposição automática de anticorpos ((condição)-(ação)), a qual representa uma das perspectivas para trabalhos futuros.

Tag	(condição)-(ação)	conexões (Idiotopos)
3 -	010—101	.0.0.0.0.0.1.1.1.1.1.1.0.0.0.0.0.1.1.0.0.0.0.0;
3 -	001—010	.0.0.0.0.0.0.0.1.0.0.0.1.0.0.1.0.0.0.0.0.0.0.0;
3 -	100—100	.0.0.0.0.0.0.0.0.1.0.0.0.1.0.0.1.0.0.0.0.0.0.0;
3 -	000—010	.0.0.0.0.0.0.0.0.0.0.0.0.0.1.1.0.1.1.0.0.0.0.0.0;
3 -	000—100	.0.0.0.0.0.0.0.0.0.0.0.0.1.1.0.1.1.0.0.0.0.0.0.0;
2 -	010—010	.0.0.0.0.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0;
2 -	010—100	.0.0.0.0.0.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0;
2 -	001—100	.0;
2 -	100—010	.0;
2 -	000—000	.0.0.0.0.0.0.0.0.0.0.1.1.1.1.1.1.1.1.0.0.0.0.1.0;
1 -	010—101	.0.1.0.0.1.0;
1 -	001—010	.0.0.0.0.0.0.0.0.0.0.1.0.0.0.0.0.0.0.0.1.0.0.1.0;
1 -	100—100	.0.0.0.0.0.0.0.0.0.0.1.0.0.0.0.0.0.0.0.1.0.0.1.0;
1 -	--- 1-111	.1.1.1.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.1.0;
1 -	001—100	.0.1.0.1;
1 -	100—010	.0.1.0.1;
1 -	010—100	.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.0.1.0.1;
1 -	010—010	.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.0.0.1.0.1;
1 -	000—000	.0.1.1.1;
3 -	--- 00000	.0.0.0.0.0.0.0.0.0.1.0.0.0.0.1.1.1.1.1.0.0.0.0.0;
3 -	--- 01000	.0.0.0.0.0.0.0.0.0.1.1.1.1.1.1.1.1.1.1.0.0.0.0.0;
3 -	--- 11000	.0.0.0.0.0.0.0.0.0.1.1.1.1.1.0.0.0.0.0.0.0.0.0.0;
4 -	--- 0-111	.1.1.1.1.1.0.0.0.0.0.0.0.0.0.1.1.1.1.1.0.0.0.0.0;
4 -	--- 1-101	.0.0.0.0.0.1.1.1.1.1.1.1.1.1.0.0.0.0.0.0.0.0.0.0;

Figura 4.6 – Codificação utilizada para a estrutura da rede imunológica projetada por WATANABE *et al.* (1999). Um número 1 na *i*-ésima posição das conexões indica que aquele anticorpo estimula o *i*-ésimo anticorpo.

Sendo assim, o trabalho se concentra na evolução das conexões, representadas pelos idiotopos. A parte conhecida como paratopo, responsável pela regra <condição>-<ação> de cada anticorpo, será mantida fixa. No Capítulo 5, em que serão apresentados os experimentos de navegação, várias propostas para o conjunto de regras <condição>-<ação> serão apresentadas, sendo que uma delas é a própria proposta de WATANABE *et al.* (1999), descrita na Figura 4.6 e resultante de um processo construtivo.

As *tags* utilizadas na Figura 4.6 referem-se a: 1-lixo; 2-obstáculo; 3-base; 4-energia interna. Alguns dos anticorpos não expressam situações relativas à direção do objeto, mas sim informações relacionadas com quantidade de energia, distância da base, e transporte ou não de lixo. Por esse motivo, apresentam codificações diferenciadas na parte que se refere à <condição>.

4.3.1 A representação

Como apresentado na Figura 4.6, para o caso específico da proposta de WATANABE *et al.* (1999), as várias possibilidades de padrões de conexão para a rede imunológica podem ser expressas na forma de uma matriz binária cujo número de elementos coincide com o quadrado do número de anticorpos da rede. A i -ésima linha da matriz representa o conjunto de idiotopos do anticorpo i . A interpretação de cada idiotopo é a seguinte: a existência de bits 1 indica a presença de conexão entre os anticorpos da linha e coluna respectivas.

Na Figura 4.4, já ficou evidenciado que cada conexão implica uma estimulação e uma supressão. Com isso, a presença de um bit 1 na linha i e coluna j implica que o anticorpo i estimula o anticorpo j e é suprimido por este.

Cada anticorpo da rede pode ter de 0 até N idiotopos, onde N é o número total de anticorpos na rede. Para a rede da Figura 4.6, utilizada no trabalho de WATANABE *et al.* (1999), o número de anticorpos é $N = 24$, e os idiotopos de cada anticorpo foram projetados manualmente, sendo que o maior número de idiotopos encontrados em um anticorpo é 10.

Na presente implementação, esse número pode variar de 0 a N . Embora fosse possível reduzir o número máximo de conexões admissíveis, foi mantido o limite de N conexões, visando admitir como candidatos à solução todos os padrões de conexão possíveis. Desse modo, não há nenhum privilégio para redes parcimoniosas, ou seja, redes que apresentam menos conexões.

O principal objetivo então é permitir que redes densamente conectadas tenham igual oportunidade de concorrer como soluções candidatas, o que seguramente não ocorre no caso de abordagens construtivas e nem mesmo quando o projetista arbitra o conjunto de conexões.

É evidente que uma rede densamente conectada dificulta sobremaneira a interpretação do resultado final, devido ao processamento distribuído. Quanto maior a quantidade de estímulos e supressões, menor a capacidade de atribuir papel específico a cada anticorpo.

Havendo a necessidade de se atribuir funcionalidades específicas a este ou aquele anticorpo, então a parcimônia pode ser incorporada junto à função de *fitness*.

4.3.2 A função de *fitness*

Cada indivíduo de uma dada geração do algoritmo evolutivo corresponde a um padrão de conexões para a rede imunológica, descrito pelos idiotopos dos anticorpos na forma de uma matriz binária. Os paratopos são constantes ao longo de todo o processo evolutivo. Esse tipo de codificação implica que cada indivíduo da população é uma possível solução para o problema.

O processo evolutivo para a síntese de um controlador autônomo é diferenciado pela complexidade de se medir o *fitness* de cada indivíduo da população. Deve-se manter um compromisso entre a qualidade da avaliação e o custo computacional.

Para isso, cada indivíduo da população, o qual corresponde a uma proposta de rede imunológica, é avaliado dentro de um tempo preestabelecido para controlar o robô em um ambiente, com as mesmas configurações e condições iniciais para todas as simulações. Nas simulações de desvio de obstáculo e captura de alvo, a função de *fitness* variou com o tipo de tarefa. A Equação (4.2) representa a função de avaliação utilizada na tarefa de coleta de lixo:

$$Fitness = \alpha N_{lixos} + \beta N_{lixo-base} + \omega N_{recargas} - \mu N_{colisões} \quad (4.2)$$

onde os termos representam o número de lixos capturados (N_{lixos}), o número de lixos transportados até a base ($N_{lixo-base}$), o número de recargas em condições de energia baixa ($N_{recargas}$), e o número de colisões ($N_{colisões}$). Os coeficientes para cada termo são determinados de forma que as melhores redes sejam aquelas que realizam o maior número de tarefas completas (localização do lixo, captura e transporte até a base) em uma simulação.

A utilização de uma função de *fitness* adequada é muito importante para o bom desempenho da evolução. No caso da Equação (4.2), a avaliação de cada indivíduo da população não será completamente satisfatória, pois uma mesma rede pode apresentar desempenhos diferentes em um mesmo ambiente. Assim, o desempenho de cada indivíduo foi medido a partir de uma média de três simulações.

A função de *fitness* apresentada na Equação (4.2) contém essencialmente o que se deseja como objetivo da navegação. Ela está de acordo, portanto, com a sugestão de NOLFI (1997), que recomenda a exclusão de termos que não retratem aspectos essenciais do processo de navegação, mesmo que a ausência desses termos específicos provoque uma maior variância na avaliação de desempenho.

Como mencionado anteriormente, uma medida de parcimônia associada ao número de conexões pode ser incorporada à função de *fitness*, caso se queira privilegiar propostas de redes imunológicas caracterizadas por um menor número de conexões, levando

possivelmente a um aumento na interpretabilidade do papel de cada anticorpo da rede imunológica.

4.3.3 Operadores Genéticos

O processo evolutivo utilizado foi dividido em duas etapas. Na primeira etapa, foram utilizados os operadores de crossover e mutação, e na segunda somente o operador de mutação. Cada etapa produziu como resultado redes imunológicas capazes de atender os objetivos da navegação.

Na primeira etapa, o algoritmo genético básico utilizado seguiu as diretrizes de GOLDBERG (1989), contendo as fases de avaliação, seleção dos pais, aplicação de crossover e mutação e a substituição na população.

Na segunda etapa, onde o operador de mutação foi aplicado isoladamente, foram utilizados conceitos de mutação observados no trabalho de NOLFI (1997) e NOLFI & FLOREANO (2000), que realizaram a evolução de controladores autônomos para robôs móveis baseados em redes neurais artificiais.

Crossover

O operador de crossover toma duas redes imunológicas da população e realiza a troca de um número específico de anticorpos, que sejam idênticos na porção do paratopo (crossover entre nós homólogos). Esses anticorpos selecionados trocarão entre si a porção do idiotopo, ou seja, o índice das conexões que realizam na rede. No decorrer do processo, as melhores configurações de conexões tendem a permanecer. Isso ocorre porque o robô, exposto às condições de navegação específicas para a avaliação, identifica os anticorpos que devem ser estimulados em uma dada situação captada pelos sensores. A rede que obtiver a melhor média em todas as ações tomadas conterá os anticorpos com estruturas mais adequadas aos objetivos da navegação.

O algoritmo de aplicação do operador é descrito a seguir. Depois de selecionadas as redes-pais pela técnica de *roulette wheel*, ou seja, levando-se em conta o *fitness*, um número aleatório de anticorpos (limitado à metade do número N de anticorpos da rede) é trocado entre elas, conforme apresentado na Figura 4.7.

A taxa ou porcentagem com que é aplicado o crossover é de 40% sobre o tamanho da população, que no caso da coleta de lixo é de 100 indivíduos na população. O operador de crossover mostrou-se adaptado ao problema e proporcionou bons resultados. Esses resultados foram apresentados em MICHELAN & VON ZUBEN (2002) e serão melhor detalhados no capítulo 5 desta dissertação.

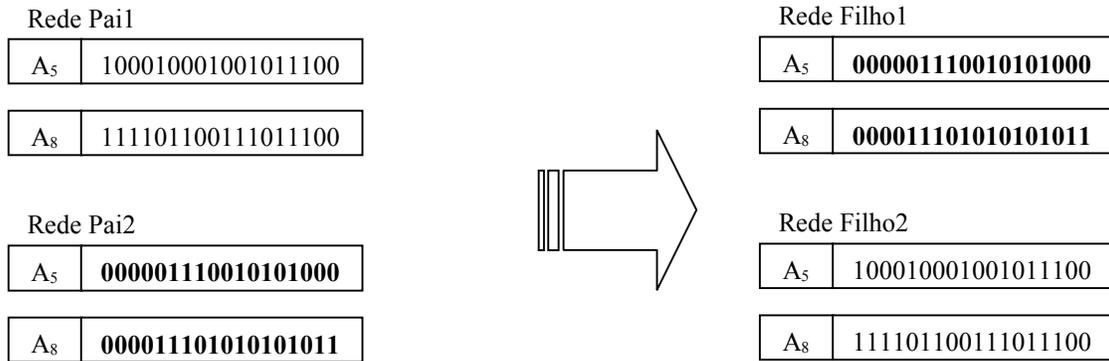


Figura 4.7 – Exemplo de aplicação do operador de crossover, onde os anticorpos A_5 e A_8 são trocados e as redes resultantes passam a ter nova configuração de conexões. Os demais anticorpos de cada rede imunológica não sofrem nenhuma alteração, razão pela qual não foram apresentados.

Mutação

O operador de mutação, em um primeiro momento, foi utilizado a fim de promover alterações nas conexões dos indivíduos criados a partir da aplicação do operador de crossover. A mutação se deu a uma taxa de 1%. Quando um indivíduo é selecionado pelo operador de mutação, um de seus anticorpos é escolhido aleatoriamente. Esse anticorpo tem então um atributo de seu idiotopo alterado, conectando ou descontando-o de outro anticorpo da rede.

A operação de mutação segue então a forma da Equação (4.3):

$$m_{ij} = 1 - m_{ij}, \quad (4.3)$$

que fixa o valor em 0 ou 1, e relaciona a afinidade m entre o anticorpo i selecionado e um j escolhido aleatoriamente. Caso a afinidade entre i e j seja 1 isso indica a existência da conexão.

Os resultados obtidos utilizando-se crossover e mutação juntos, exibidos em MICHELAN & VON ZUBEN (2002), foram animadores, visto que os objetivos da navegação foram atendidos. Além disso, quando comparado à rede proposta por WATANABE *et al.* (1999), o desempenho foi equivalente. Maiores detalhes sobre esses resultados também fazem parte do capítulo 5.

Mutação como único operador

Na segunda etapa de implementação dos processos evolutivos, a mutação foi aplicada como único operador genético na evolução, baseado em resultados obtidos por NOLFI (1997) e NOLFI & FLOREANO (2000).

Nesse modelo, os 20% melhores indivíduos da população são selecionados e geram 4 cópias cada. As cópias somadas aos seus pais constituirão a nova geração. Assim, em

uma população de 100 redes, 20 são selecionadas e para cada rede são feitas quatro cópias que sofrerão mutação à taxa de 20%. Desse modo, a população mantém seu tamanho original substituindo os outros 80 indivíduos pelos filhos gerados.

Para os 20% que sofrem mutação, a escolha do valor é feita de maneira aleatória: ou será 0 ou será 1. Isso indica que, apesar de selecionado, o bit que indica uma ligação ativa ou não pode não ter seu valor alterado. Com isso, a porcentagem de bits efetivamente alterados varia de 0 a 20%.

Método de Seleção

Espera-se que o sistema seja capaz de encontrar, através do processo evolutivo, as estruturas de redes que sejam mais adaptadas aos objetivos da navegação. Para tanto, o método de seleção elitista foi utilizado e, assim, as redes com os maiores valores de fitness são mantidas, e as redes com os piores valores de fitness são substituídas a certa taxa, dando lugar aos novos indivíduos gerados.

Na primeira etapa do processo evolutivo, em que foram utilizados os operadores de crossover e de mutação, a taxa de substituição é de 40%. Na segunda etapa, quando se passou a utilizar a mutação como único operador genético, a taxa saltou para 80%.

4.4 Ambiente de simulação

Para a avaliação da população composta por redes imunológicas, foi desenvolvido um ambiente de simulação reconfigurável que proporciona todos os recursos necessários, levando em consideração os problemas enfrentados em navegação autônoma, e também provendo as características associadas ao problema de coleta de lixo em questão (veja Figura 4.8). O ambiente foi modelado levando em consideração conceitos de orientação a objeto, o que propicia o reaproveitamento do código e teste de outros sistemas de controle. Maiores detalhes sobre as características do ambiente são encontrados no Apêndice A.

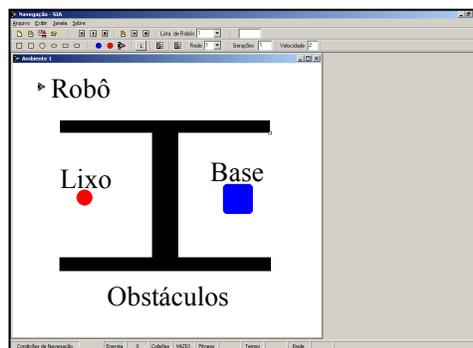


Figura 4.8 – O ambiente provê estruturas que representam lixo (círculos vermelhos), bases (retângulos azuis de cantos arredondados) e obstáculos (objetos em preto).

É importante lembrar que a simulação constitui um passo muito importante na concepção de sistemas de controle, principalmente em navegação, onde há custos para a implantação e teste em robôs reais. No capítulo 5, serão discutidos os resultados obtidos em simulação computacional e em alguns experimentos preliminares com um robô real.

4.5 Arquitetura do robô

O robô modelado para o ambiente de simulação tem como objetivo reproduzir as características do robô real (Khepera II), no qual foram realizados os experimentos. Para isso, a arquitetura do robô foi construída para contar com recursos equivalentes aos disponíveis no Khepera II e que são necessários para o sistema em estudo. O modelo computacional do robô é descrito no Apêndice A e as características técnicas mais relevantes, assim como aspectos da implantação do sistema proposto no robô Khepera II, são encontradas no Apêndice B.

Apesar do robô Khepera II já contar com um software de simulação, disponível no mercado, o desenvolvimento de um ambiente de simulação customizado se mostrou plenamente justificável, pois proporcionou maior domínio das configurações e extensões necessárias. Foi feito um investimento em interfaces responsáveis pela exibição de resultados e outras informações necessárias para uma boa análise das redes obtidas via algoritmos evolutivos.

Capítulo 5

Resultados e Discussões

Neste capítulo, descrevem-se inicialmente os resultados alcançados via simulação computacional, seguidos pelos primeiros experimentos práticos, realizados com o robô real (Khepera II). Os resultados foram obtidos utilizando o ambiente de simulação desenvolvido e adaptado às características do problema em questão.

Os resultados exibidos na Seção 5.1 têm por objetivo demonstrar a viabilidade da aplicação do processo evolutivo na concepção de estruturas de redes imunológicas artificiais, fazendo uma comparação entre o desempenho de uma das redes obtidas e a estrutura de rede proposta em WATANABE *et al.* (1999). A estrutura da rede evoluída e utilizada na simulação é posteriormente comparada com a estrutura proposta, a fim de se verificar que, apesar do alto número de conexões, o objetivo é alcançado (MICHELAN & VON ZUBEN, 2002).

Na Seção 5.2, são exibidos os resultados obtidos com a alteração no algoritmo evolutivo, aplicando-se somente a mutação como operador. Nas duas primeiras simulações, Seções 5.2.1 e 5.2.2, as tarefas do robô são alteradas, visando os experimentos práticos com o robô Khepera II. Novos repertórios de anticorpos para as tarefas modificadas foram criados. Para cada uma das tarefas, desvio de obstáculos, desvio de obstáculos e captura de alvo e coleta de lixo, foram evoluídas redes imunológicas específicas, e os resultados de simulação indicam a capacidade das redes obtidas em realizar a tarefa de navegação.

Na Seção 5.3, são exibidos os resultados preliminares obtidos em experimentos práticos com a implantação do sistema de controle proposto no robô Khepera II. Nas Seções 5.3.1, 5.3.2 e 5.3.3, a tarefa experimentada é a de desvio de obstáculos, onde se espera que a rede imunológica evoluída e utilizada nas simulações da Seção 5.2.1, que controla o robô, seja capaz de demonstrar comportamentos como tangenciar e desviar de obstáculos. Na Seção 5.3.4, é exibido o acompanhamento do nível de concentração dos anticorpos, repetindo o experimento da Seção 5.3.3.

Em todos os testes realizados, o aspecto mais importante a ser ressaltado, é a capacidade de coordenação exibida pelas redes imunológicas obtidas através do processo evolutivo. A tarefa de concepção manual das conexões é uma tarefa complexa e, diante do

aumento do número de anticorpos, inviável mesmo para um especialista, de maneira que a construção automática se faz necessária.

Um ponto importante a ser considerado é a característica exploratória do robô. Enquanto nenhum evento significativo é detectado, a navegação tende a ser errática. No entanto, a partir do momento em que o robô localiza um obstáculo, um lixo ou uma base, os quais podem ser considerados como alvos da navegação, o comportamento de navegação deve convergir para o atendimento dos objetivos.

5.1 Evolução via Crossover e Mutação

Inicialmente, serão descritos os resultados alcançados na primeira etapa dos experimentos, levando em consideração a aplicação conjunta dos operadores de crossover e mutação. Os resultados de simulação obtidos a partir da estrutura da rede resultante do processo evolutivo serão comparados com a rede proposta por WATANABE *et al.* (1999).

O número de indivíduos da população é de 100 redes, compostas de 24 anticorpos, concebidos para a tarefa de coleta de lixo, ou seja, visando sub-tarefas como desvio de obstáculos, captura de lixos e bases. As conexões da população são iniciadas aleatoriamente. A estrutura de <condição>-<ação> de cada anticorpo é exibida a seguir. A codificação desses anticorpos, assim como suas conexões, são exibidas na Figura 4.6.

1-<Lixo à frente>-<Vá em frente>	13-<Nenhuma base>-<Vire à direita>
2-<Lixo à direita>-<Vire à direita>	14-<Nenhuma base>-<Vire à esquerda>
3-<Lixo à esquerda>-<Vire à esquerda>	15-<Base muito longe>-<Explore>
4-<Carregando Lixo>-<Procure a base>	16-<Base longe>-<Explore>
5-<Lixo à direita>-<Vire à esquerda>	17-<Base perto>-<Explore>
6-<Lixo à esquerda>-<Vire à direita>	18-<Obst. à frente>-<Vire à direita>
7-<Lixo à frente>-<Vire à esquerda>	19-<Obst. à frente>-<Vire à esquerda>
8-<Lixo à frente>-<Vire à direita>	20-<Obst. à direita>-<Vire à esquerda>
9-<Nenhum Lixo>-<Vire à direita>	21-<Obst. à esquerda>-<Vire à direita>
10-<Base à frente>-<Vá em frente>	22-<Nenhum obs>-<Explore>
11-<Base à direita>-<Vire à direita>	23-<Energia baixa>-<Procure a base>
12-<Base à esquerda>-<Vire à esquerda>	24-<Energia alta>-<Vá em frente>

No conjunto de anticorpos apresentados, observa-se em alguns casos comportamentos antagônicos para a mesma condição. Isso se deve ao fato de que a tarefa escolhida possui mais que um objetivo. Repare que, em certas condições, o robô está

buscando o lixo e, por isso, deve persegui-lo, tome como exemplo o anticorpo 1. Agora, suponha que o robô já capturou o lixo e, nesse caso, está em busca da base. Dessa maneira, ele não deve mais direcionar-se para o lixo que estiver à sua frente. Então tome os anticorpos 7 e 8 como exemplos: é função do conjunto de anticorpos fornecer meios para que a tarefa de desvio do lixo possa ser cumprida com êxito pelo robô. O antagonismo nessas situações é resolvido pela própria dinâmica da rede, que altera as concentrações dos anticorpos de acordo com a situação corrente, ou seja, de acordo com os antígenos.

Cada um dos indivíduos foi avaliado no mesmo ambiente com as mesmas condições iniciais e o *fitness* foi atribuído segundo a Equação (4.2), descrita na Seção 4.3.2.

$$Fitness = \alpha N_{lixos} + \beta N_{lixo-base} + \omega N_{recargas} - \mu N_{colisões} \quad (4.2)$$

Em função das características do controle realizado, uma mesma rede pode ter diferentes trajetórias em um mesmo ambiente. Lembre-se que a ação a ser adotada a cada instante é escolhida aleatoriamente, sendo que a probabilidade de cada anticorpo indicar a ação é proporcional à sua concentração naquele momento, empregando-se o método da roleta (*roulette wheel*). Devido a essa variabilidade de comportamento, o *fitness* de cada um dos indivíduos é uma média de desempenho de três simulações.

Os operadores genéticos de crossover e mutação foram aplicados a taxas de 40% e 1%, respectivamente. As redes substituídas em cada geração foram selecionadas via elitismo à taxa de 40%. Foi estipulada a condição de parada de 500 gerações. Tal condição foi arbitrada a partir de exaustivos testes e está diretamente relacionada com a complexidade da tarefa a ser executada pelo robô. Após a conclusão do processo evolutivo, a rede com maior *fitness* foi utilizada para produzir os resultados de simulação apresentados na Figura 5.2.

A Figura 5.1 apresenta os resultados obtidos utilizando a estrutura de rede proposta por WATANABE *et al.* (1999).

Os pontos vermelhos correspondem aos lixos e o retângulo azul à base. A linha descreve a trajetória percorrida pelo robô. Já os pontos descrevem os lugares onde houve contato do robô ao longo do percurso. Estes pontos de contato não estão encostados nos respectivos módulos de contato simplesmente porque se está plotando o centro de massa do robô, quando o robô apresenta uma dimensão não-desprezível. Além disso, a colisão está sendo indicada pela existência de uma distância, entre o robô e algum objeto, abaixo de um determinado limiar, sendo que este limiar é maior que zero.

O objetivo nas simulações das Figuras 5.1 e 5.2 é comprovar a capacidade das estruturas de redes imunológicas apresentadas no tocante à coordenação dos módulos de comportamento elementar para a realização da tarefa de coleta de lixo, mantendo um nível mínimo de energia interna.

No caso da Figura 5.1, o desempenho foi satisfatório, pois o robô realizou sequencialmente a tarefa de coletar lixo e levá-lo até a base, mantendo a energia interna.

Observa-se que o robô realizou três coletas e apenas uma entrega na base. Neste caso, não há nenhum obstáculo interno no ambiente de navegação.

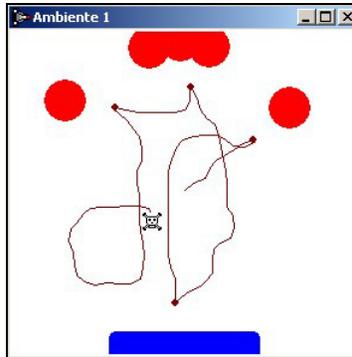


Figura 5.1 – Simulação com a rede proposta por WATANABE *et al.* (1999).

A rede evoluída segundo a proposta desta dissertação foi testada no mesmo ambiente, com as mesmas condições iniciais. Ela demonstrou capacidade de coordenação para a realização da tarefa. O robô capturou o lixo e levou-o à base por duas vezes e antes do tempo limite ainda realizou mais uma captura de lixo, conforme indicado na Figura 5.2.

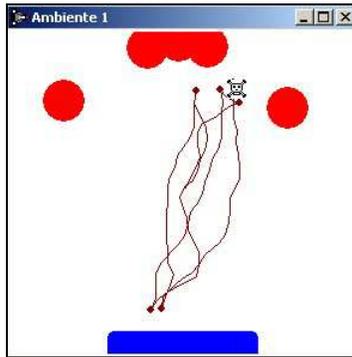


Figura 5.2 – Simulação com a rede evoluída utilizando os dois operadores genéticos.

É importante salientar que os robôs, em ambas as simulações, atingiram o tempo limite de simulação mantendo o nível de energia interna acima do limiar. A arquitetura das redes imunológicas é apresentada na Figura 5.3. Nelas, é possível observar que a rede evoluída (veja Figura 5.3B) possui um número muito maior de conexões.

Não houve nenhuma preocupação em minimizar o número de conexões ao longo do processo evolutivo. Isto implica que a interpretabilidade da rede resultante do processo evolutivo fica comprometida, mas isto também demonstra que existem propostas bem sucedidas de padrões de conexão que diferem significativamente daquela proposta por um especialista. Outra observação importante diz respeito à maneira como foram plotadas as estruturas das redes. Não há interpretação espacial, mas apenas uma forma de explicitar as afinidades entre os anticorpos.

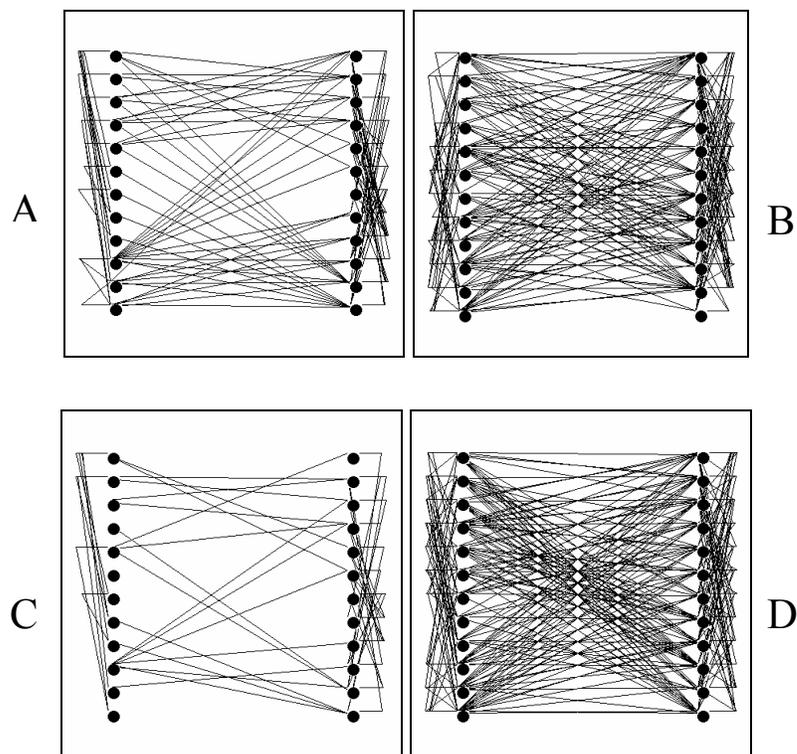


Figura 5.3 – Estruturas das redes utilizadas nas simulações. A estrutura A representa a rede proposta no trabalho de WATANABE *et al.* (1999), com 100 conexões, e a estrutura B a rede evoluída pelo sistema imuno-genético, com 231 conexões. A estrutura C representa as 47 conexões existentes simultaneamente nas redes A e B, e a estrutura da figura D as 237 conexões diferentes nas duas redes.

Quando são comparadas as estruturas das redes, a proposta por WATANABE *et al.* (1999) (veja Figura 5.3A) e a evoluída (veja Figura 5.3B) possuem 47 conexões comuns, exibidas na estrutura da Figura 5.3C, e 237 conexões diferentes, apresentadas na Figura 5.3D.

5.2 Evolução via mutação

Na segunda etapa do trabalho, foi aplicado como único operador genético a mutação, a fim de se observar o desempenho da evolução. A reprodução dos indivíduos mais adaptados é realizada através de cópias, seguida de aplicação da mutação a alguns bits selecionados aleatoriamente (Seção 4.3.2). Esse processo permitiu melhores resultados quando comparado à aplicação de crossover e mutação juntos. Isso se deve ao fato de que os nós da rede são intimamente dependentes. Resultados semelhantes foram relatados em NOLFI (1997) quando da aplicação deste processo evolutivo na busca de vetores de pesos para redes neurais artificiais.

Os conceitos relacionados à função de *fitness* empregada foram baseados em relatos de NOLFI (1997) e NOLFI & FLOREANO (2000). Os indivíduos são avaliados de maneira que

aqueles que demonstram algum tipo de qualidade desejável recebam uma avaliação superior àqueles menos adaptados.

5.2.1 Evolução via mutação: tarefa de desvio de obstáculos

Visando os testes iniciais com o robô real, foram feitas algumas mudanças a fim de facilitar a transferência do sistema de controle para o ambiente real e iniciar a validação da rede evoluída como sistema de controle autônomo. A principal mudança foi a modificação dos objetivos a serem cumpridos pelo robô. Para tanto, o conjunto de anticorpos, responsáveis pelos pares de <condição>-<ação>, também foi modificado. Um repertório de 13 anticorpos, exibido a seguir, foi criado com a finalidade de fornecer ao robô condições seguras de navegação, desde que coordenados adequadamente.

Regras	Codificação	
1-<Obstáculo à frente, perto>-<Vire à direita, forte>	2 010	11 010
2-<Obstáculo à frente, longe>-<Vire à direita, normal>	2 010	10 001
3-<Obstáculo à frente, muito longe>-<Vá em frente>	2 010	01 101
4-<Obstáculo à frente, perto>-<Vire à esquerda, forte>	2 010	11 100
5-<Obstáculo à frente, longe>-<Vire à esquerda, normal>	2 010	10 110
6-<Obstáculo à frente, muito longe>-<Vá em frente>	2 010	01 101
7-<Obstáculo à esquerda, muito longe>-<Vire à direita, fraco>	2 100	01 000
8-<Obstáculo à esquerda, longe>-<Vire à direita, normal>	2 100	10 001
9-<Obstáculo à esquerda, perto>-<Vire à direita, forte>	2 100	11 010
10-<Obstáculo à direita, muito longe>-<Vire à esquerda, fraco>	2 001	01 011
11-<Obstáculo à direita, longe>-<Vire à esquerda, normal>	2 001	10 110
12-<Obstáculo à direita, perto>-<Vire à esquerda, forte>	2 001	11 100
13-<Nenhum Obstáculo>-<Vá em frente>	2 000	00 101

Para se alcançar êxito na transferência do sistema de controle de coleta de lixo, iniciaram-se os experimentos com tarefas simplificadas, levando em consideração que coletar lixo é a composição de sub-tarefas como desviar de obstáculos, alcançar a base e capturar lixos, sendo esses dois últimos relacionados à captura de alvos. Com isso a transferência do sistema de controle será gradual e os resultados desses experimentos serão mostrados mais adiante.

Dadas essas alterações, foram evoluídas 100 redes com os novos anticorpos, específicos para a tarefa de desvio de obstáculos, durante 200 gerações. Apesar de parecer uma tarefa simples, ela exige muito do módulo de coordenação, principalmente em situações de conflito como, por exemplo, quando os sensores captam obstáculos em mais de uma direção e o robô tem que optar por se distanciar daquele que estiver mais próximo,

ou no dilema do “beco”, onde o robô se vê cercado de obstáculos por todos os lados. Para essa tarefa, a função de *fitness* apresentada na Equação (5.2) leva em consideração a capacidade de desvio. Assim, os indivíduos que colidirem menos são os mais recompensados. O valor final de *fitness* para cada indivíduo é uma média de três simulações.

$$Fitness = \begin{cases} 1 & \text{Se } N_{colisões} \geq 10 \\ 200 & \text{Se } N_{colisões} < 10 \text{ e } N_{colisões} \geq 7 \\ 500 & \text{Se } N_{colisões} < 7 \text{ e } N_{colisões} \geq 5 \\ 800 & \text{Se } N_{colisões} < 5 \text{ e } N_{colisões} \geq 3 \\ 1500 & \text{Se } N_{colisões} < 3 \text{ e } N_{colisões} \geq 2 \\ 3500 & \text{Se } N_{colisões} < 2 \end{cases} \quad (5.2)$$

Uma rede construída com o conjunto de anticorpos utilizado para a evolução, mas sem possuir nenhuma conexão (interação dos anticorpos), foi testada no mesmo ambiente da rede obtida via evolução. A Figura 5.4 apresenta o desempenho dessa rede dentro do tempo limite estipulado, onde se percebe o grande número de colisões (pontos na linha da trajetória) e a ausência de movimentos coordenados. Essa simulação tem o objetivo de demonstrar a importância da interação dos anticorpos na definição do nível de concentração de cada um. Pode-se interpretar o nível de concentração dos anticorpos como uma memória das ações e informações sensoriais já processadas. Isso justifica o empenho para se encontrar a estrutura com as conexões adequadas, através do processo evolutivo.

Quando a rede evoluída é testada nas mesmas condições, percebe-se que os movimentos são coordenados e as decisões tomadas são adequadas para as situações encontradas. A Figura 5.5 exibe o resultado dessa simulação, onde o robô conseguiu manter sua integridade não colidindo durante o tempo de navegação.

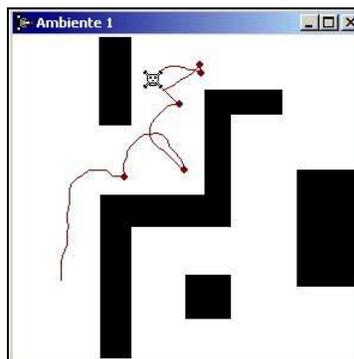


Figura 5.4 – Ausência de coordenação nas decisões tomadas: o robô não consegue navegar sem colidir.

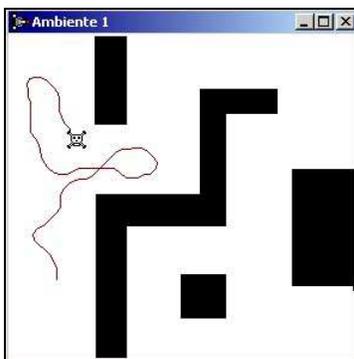


Figura 5.5 – Presença de coordenação nas decisões tomadas. Verifica-se que foram tomadas decisões adequadas durante a navegação, fazendo com que o robô se desvie dos obstáculos encontrados.

Por último, na Figura 5.6, a rede mostrou-se bastante eficiente na coordenação, mesmo com maior velocidade, explorando grande parte do ambiente. Componentes do sistema, como a velocidade de translação e de rotação, são importantes e devem ser ajustados às características do ambiente e da tarefa.

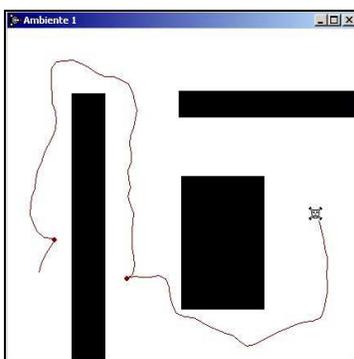


Figura 5.6 – Simulação de desvio de obstáculo com velocidade maior que a utilizada na Figura 5.5. O robô conseguiu navegar com apenas duas colisões.

A estrutura da rede utilizada nas simulações (veja Figura 5.7A) apresenta a mesma característica de elevado número de conexões. Quando comparada com a estrutura de outra rede evoluída (veja Figura 5.7B) e igualmente capaz de realizar a tarefa, observa-se quase o mesmo panorama. A estrutura da Figura 5.7C representa as 44 conexões iguais presentes nas redes A e B. As conexões diferentes da Figura 5.7D são em número de 83. Se observarmos que o número de conexões da rede A é 78 e o da rede B é 93, verifica-se que o número de conexões iguais é aproximadamente 50% do número de conexões totais em cada rede.

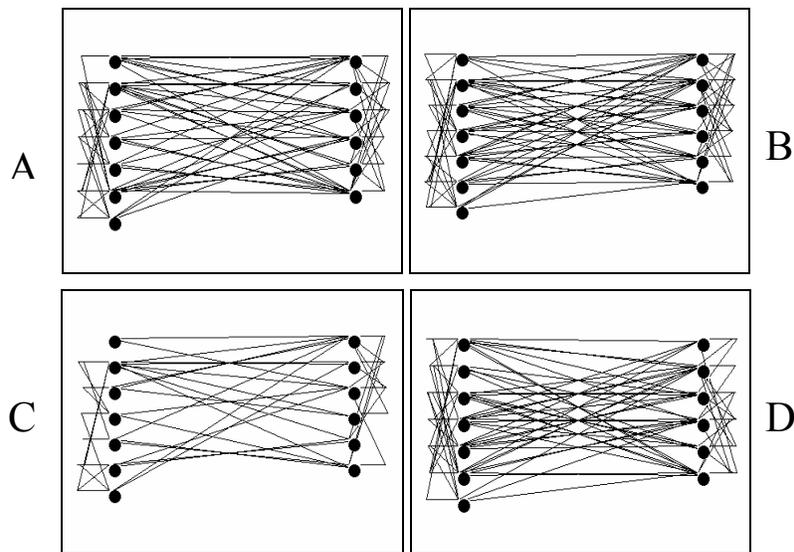


Figura 5.7 – Estruturas das redes utilizadas nas simulações. As estruturas A e B representam as redes evoluídas pelo sistema imuno-genético, onde A possui 78 conexões e B 93 conexões. A estrutura C representa as 44 conexões iguais existentes nas redes A e B, e a estrutura da figura D as 83 conexões diferentes nas duas redes.

Diante das simulações apresentadas, é nítida a capacidade da rede imunológica como mecanismo de coordenação. Do mesmo modo, a evolução de redes demonstrou ser viável e dessa forma automatizou-se uma fase do processo de concepção do sistema de controle.

5.2.2 Evolução via mutação: tarefa de desvio de obstáculos e captura de alvo

Assim como as redes para a tarefa de desvio de obstáculos, as redes evoluídas para a tarefa de desvio de obstáculos e captura de alvo tiveram seu conjunto de anticorpos alterado. Foi criado um repertório de 18 anticorpos que fornecem ao robô condições de navegar e capturar o alvo. Esses anticorpos e suas codificações são exibidos a seguir.

Regras	Codificação
1-⟨Obstáculo à frente, perto⟩-⟨Vire à direita, forte⟩	2 010 11 010
2-⟨Obstáculo à frente, longe⟩-⟨Vire à direita, normal⟩	2 010 10 001
3-⟨Obstáculo à frente, muito longe⟩-⟨Vá em frente⟩	2 010 01 101
4-⟨Obstáculo à frente, perto⟩-⟨Vire à esquerda, forte⟩	2 010 11 100
5-⟨Obstáculo à frente, longe⟩-⟨Vire à esquerda, normal⟩	2 010 10 110
6-⟨Obstáculo à frente, muito longe⟩-⟨Vá em frente⟩	2 010 01 101
7-⟨Obstáculo à esquerda, muito longe⟩-⟨Vire à direita, fraco⟩	2 100 01 000

8-⟨Obstáculo à esquerda, longe⟩-⟨Vire à direita, normal⟩	2 100	10 001
9-⟨Obstáculo à esquerda, perto⟩-⟨Vire à direita, forte⟩	2 100	11 010
10-⟨Obstáculo à direita, muito longe⟩-⟨Vire à esquerda, fraco⟩	2 001	01 011
11-⟨Obstáculo à direita, longe⟩-⟨Vire à esquerda, normal⟩	2 001	10 110
12-⟨Obstáculo à direita, perto⟩-⟨Vire à esquerda, forte⟩	2 001	11 100
13-⟨Nenhum Obstáculo⟩-⟨Vá em frente⟩	2 000	00 101
14-⟨Alvo à frente⟩-⟨Vá em frente⟩	3 010	-- 101
15-⟨Alvo à direita⟩-⟨Vire à direita, forte⟩	3 001	-- 010
16-⟨Alvo à esquerda⟩-⟨Vire à esquerda, forte⟩	3 100	-- 100
17-⟨Nenhum Alvo⟩-⟨Vire à direita, forte⟩	3 000	-- 010
18-⟨Nenhum Alvo⟩-⟨Vire à esquerda, forte⟩	3 000	-- 100

Descartou-se a possibilidade de iniciar o processo evolutivo tendo como base as redes obtidas na evolução para a tarefa de desvio de obstáculos. Isso porque a evolução leva em consideração o conjunto dos anticorpos. Sendo assim, mesmo que se utilizasse uma população já evoluída para desvio de obstáculos, suas conexões seriam alteradas de forma a buscar uma nova configuração. O ambiente utilizado na evolução é diferente do apresentado aqui, o que demonstra que não houve memorização da trajetória. Na seqüência de simulações, ficará evidente que o sistema é capaz de se adaptar a mudanças no ambiente. Nessa tarefa, a função de *fitness* apresentada na Equação (5.3) privilegia os indivíduos que colidem menos e que capturam o alvo (quadrado azul).

$$\begin{aligned}
 X &= \begin{cases} 1 & \text{Se } N_{\text{colisões}} \geq 10 \\ 200 & \text{Se } N_{\text{colisões}} < 10 \text{ e } N_{\text{colisões}} \geq 7 \\ 500 & \text{Se } N_{\text{colisões}} < 7 \text{ e } N_{\text{colisões}} \geq 5 \\ 800 & \text{Se } N_{\text{colisões}} < 5 \text{ e } N_{\text{colisões}} \geq 3 \\ 1500 & \text{Se } N_{\text{colisões}} < 3 \text{ e } N_{\text{colisões}} \geq 2 \\ 3500 & \text{Se } N_{\text{colisões}} < 2 \end{cases} \\
 k &= \begin{cases} 1 & \text{Se } N_{\text{captura}} \geq 1 \\ 0 & \text{Se } N_{\text{captura}} = 0 \end{cases}
 \end{aligned} \tag{5.3}$$

$$\textit{Fitness} = X + (X * (k * 0,5))$$

A população de 100 redes foi evoluída durante 300 gerações, e a rede com o melhor *fitness* na última geração foi utilizada para as simulações a seguir.

Nesse primeiro teste, o robô foi colocado em um corredor com tempo suficiente para alcançar o alvo. A distância limite com que os sensores conseguem captar objetos é pouco menor que o tamanho do ambiente, fazendo com que o robô possa ser estimulado pelo alvo desde sua posição inicial.

Como no exemplo anterior (Seção 5.2.1), uma rede contendo o conjunto de anticorpos a serem evoluídos, mas sem nenhuma conexão entre eles, foi avaliada com o intuito de demonstrar que não há capacidade de coordenação nessa condição. A Figura 5.8 exibe os resultados dessa simulação, onde houve colisões e falta de atração do robô pelo alvo.

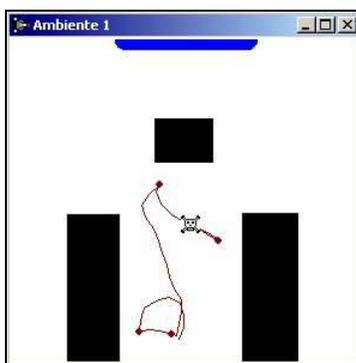


Figura 5.8 – A rede sem conexões não foi capaz de coordenar as ações adequadamente, o que implicou em um grande número de colisões. Verifica-se também a ausência de atração pelo alvo.

Ao contrário, na simulação seguinte com a rede evoluída, ficou evidente a capacidade de coordenação e a atração provocada pela visão do alvo. A tarefa foi cumprida com êxito sem colisões, conforme apresentado na Figura 5.9.

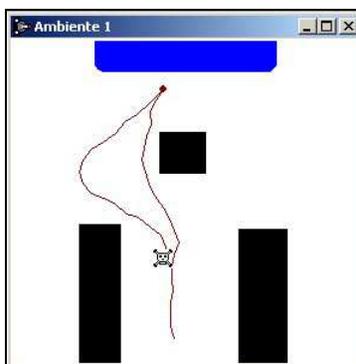


Figura 5.9 – O robô alcança o alvo desviando corretamente dos obstáculos.

Para comprovar a capacidade de generalização e desvio de obstáculos em busca do alvo, o ambiente foi alterado de forma a dificultar o acesso do robô ao alvo. Na primeira tentativa de superar os obstáculos, o robô colidiu, como mostra a Figura 5.10. Após a

colisão diante da percepção dos obstáculos o robô muda de sentido direcionando-se mais uma vez para o alvo e dessa vez supera os obstáculos.

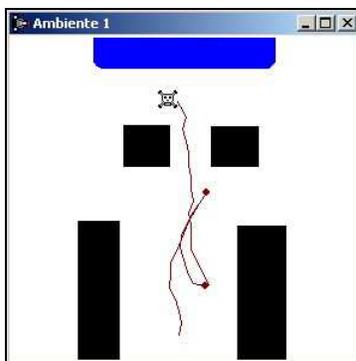


Figura 5.10 – O robô colide em sua primeira tentativa de alcançar o alvo, mas é bem sucedido na segunda superando os obstáculos.

Em outra simulação (veja Figura 5.11) com maior número de obstáculos o robô obteve sucesso e também alcançou o alvo.

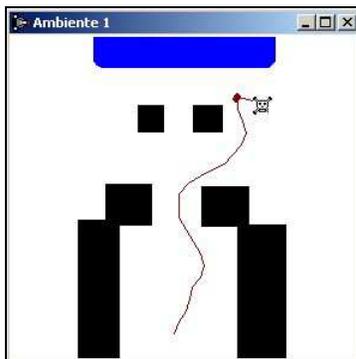


Figura 5.11 – O robô alcança o alvo mesmo com um maior número de obstáculos.

Para demonstrar o caráter exploratório do robô, foi criado um ambiente em que não fosse possível a estimulação do robô pelo alvo desde o início da simulação (veja Figura 5.12). Nota-se que, ao sair pela passagem que divide os dois ambientes, o robô localizou o alvo e foi em sua direção. Contudo, após a captura o robô explorou o ambiente navegando de forma errática, pois já não era mais atraído pelo alvo. Essa situação foi verificada em simulações diferentes, como apresentado na Figura 5.13.

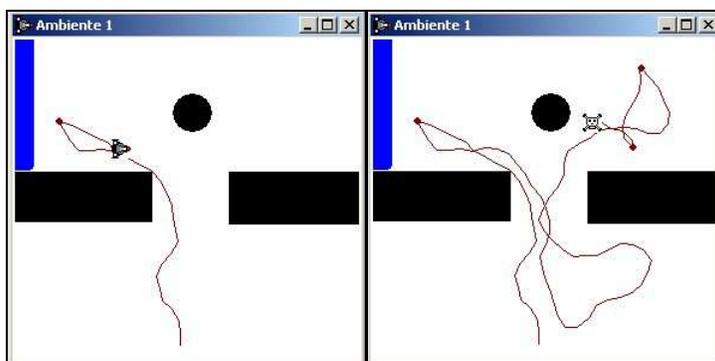


Figura 5.12 – O robô explora o ambiente, captura o alvo e depois segue explorando. O quadro à esquerda mostra apenas uma parte da trajetória completa apresentada no quadro à direita.

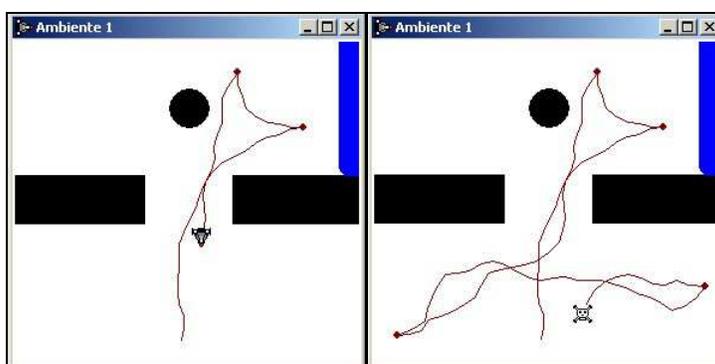


Figura 5.13 – Da mesma forma que na simulação anterior, o robô explora o ambiente, captura o alvo e, depois, segue explorando. O quadro à esquerda mostra apenas uma parte da trajetória completa apresentada no quadro à direita.

A Figura 5.14 exibe outra simulação em que o robô parte de uma área isolada sem o estímulo do alvo. Apesar dessa condição, o robô consegue sair sem colidir e ir em busca do alvo, capturando-o.

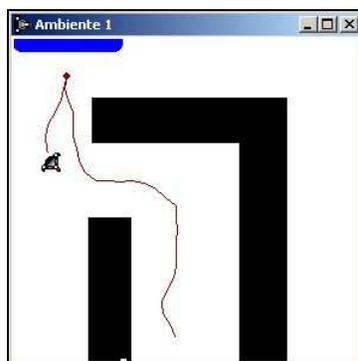


Figura 5.14 – O robô consegue sair do corredor sem colidir e realiza a captura do alvo.

Em outra simulação (veja Figura 5.15), o robô foi submetido a um ambiente em que o alvo se movia para pontos do ambiente não deixando que o robô o alcançasse. O objetivo é mostrar a capacidade de coordenação da rede em ambientes dinâmicos.

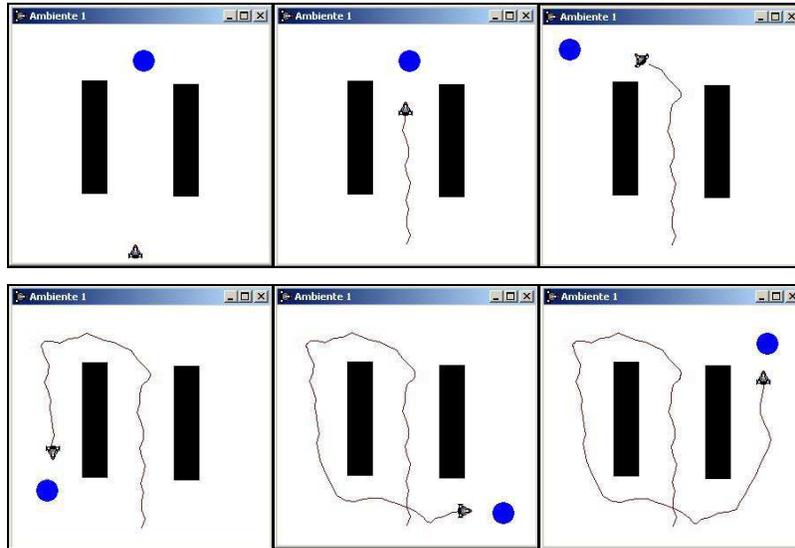
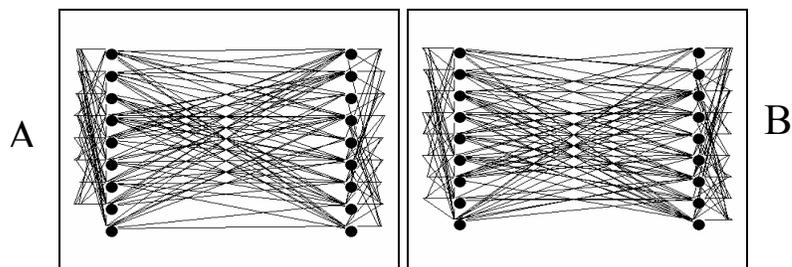


Figura 5.15 – A seqüência mostra o robô em busca do alvo, que se move para evitar que o robô o alcance.

Quando as estruturas de duas redes evoluídas (veja Figuras 5.16 A e B) e utilizadas nas simulações são comparadas, o cenário é o seguinte. A Figura 5.16C exibe as 73 conexões presentes nas duas redes. Este número, como na comparação das estruturas de rede obtidas para desvio de obstáculos, corresponde a aproximadamente 50% do número total de conexões das redes A e B, com 149 e 158 conexões, respectivamente. Contudo, fica evidente a possibilidade de diversas configurações, uma vez que existem 161 conexões diferentes entre elas.



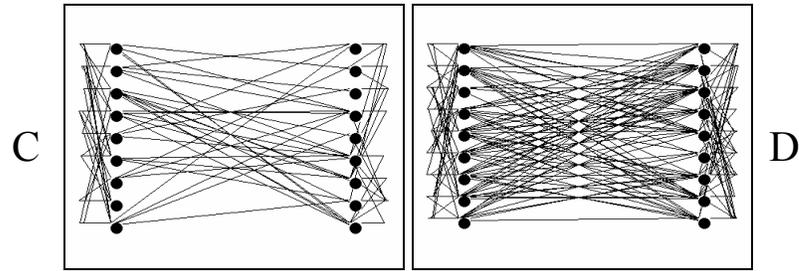


Figura 5.16 – Estruturas das redes utilizadas nas simulações. As estruturas A e B representam as redes evoluídas pelo sistema imuno-genético, onde A possui 149 conexões e B 158 conexões. A estrutura C representa as 73 conexões iguais existentes nas redes A e B, e a estrutura da figura D as 161 conexões diferentes nas duas redes.

Como mostram os resultados das simulações, as redes evoluídas também se mostraram aptas a realizar a tarefa a que foram designadas. Observa-se também comportamentos como o tangenciamento e desvio de obstáculos, assim como a busca do alvo.

5.2.3 Evolução via mutação: tarefa de coleta de lixo

Nas simulações apresentadas a seguir, o robô tem o mesmo objetivo das situações apresentadas na Seção 5.1, a coleta de lixo. O desempenho da rede evoluída será comparado ao desempenho obtido pela estrutura de rede proposta por WATANABE *et al.* (1999).

A evolução, neste caso, adota como função de *fitness* a Equação (5.4), onde são estabelecidos prêmios para os indivíduos que concluírem metade da tarefa (capturar o lixo), tarefa inteira (capturar o lixo e levá-lo à base; ou recarregar as baterias em caso de energia baixa), sendo que as ações indesejadas são fortemente penalizadas pela função. A população é constituída de 100 redes, evoluídas por 500 gerações.

$$Fitness = \alpha N_{lixos} + \beta N_{lixo-base} + \omega N_{recargas} - \mu N_{colisões} - \chi N_{lixo-lixo} - \sigma N_{base-EnergiaAlta} - \delta N_{lixo-EnergiaBaixa} \quad (5.4)$$

onde:

N_{lixos} : é o número de lixos capturados;

$N_{lixo-base}$: é o número de lixos capturados e entregues à base;

$N_{recargas}$: é o número de recargas quando a energia está baixa;

$N_{colisões}$: é o número de colisões;

$N_{lixo-lixo}$: é o número de lixos capturados quando o robô ainda não entregou o lixo que está carregando;

$N_{base-EnergiaAlta}$: é o número de vezes em que, com energia alta, o robô dirigiu-se para a base;

$N_{base-EnergiaBaixa}$: é o número de vezes em que, com energia baixa, o robô capturou lixo.

Serão apresentadas, a seguir, simulações em que as redes são expostas a situações conflitantes e, assim, pode-se avaliar a capacidade de decisão do sistema.

A primeira situação conflitante ocorre quando o robô, estando com sua energia interna baixa, encontra um lixo à frente e uma base à sua direita. Essa situação foi dada como exemplo na Seção 4.2.5. Neste caso, o robô deve atender à sua necessidade de recarregar as baterias e, assim, ignorar o lixo.

A seqüência de figuras mostra que a rede tomou a atitude esperada e levou o robô até a base, onde foi feita a recarga das baterias (veja Figura 5.17).

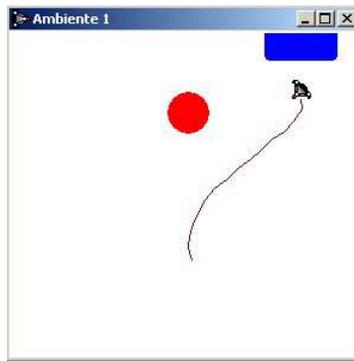


Figura 5.17 – O robô ignora o lixo, buscando a base e recarregando suas baterias.

Após a recarga e assim com energia suficiente, o robô captura o lixo e segue então novamente para a base para depositá-lo (veja Figura 5.18).

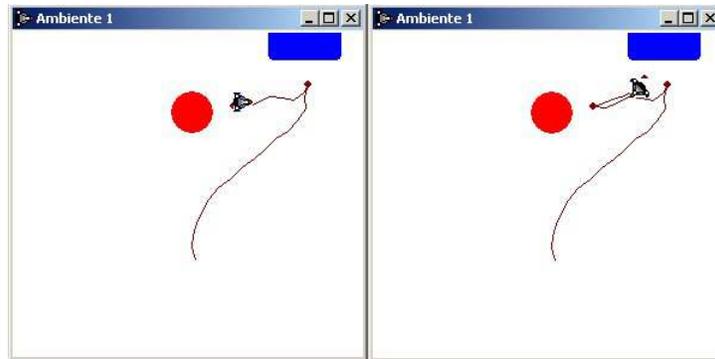


Figura 5.18 – O robô captura o lixo, buscando a base, depositando o lixo e recarregando suas baterias.

Seguindo o objetivo da navegação, o robô volta a capturar o lixo, mas dessa vez não é estimulado pela base logo após a captura, o que o faz explorar o ambiente. Nessa exploração, o robô encontra novamente o lixo, mas desvia dele, pois já está carregando um lixo (veja Figura 5.19).

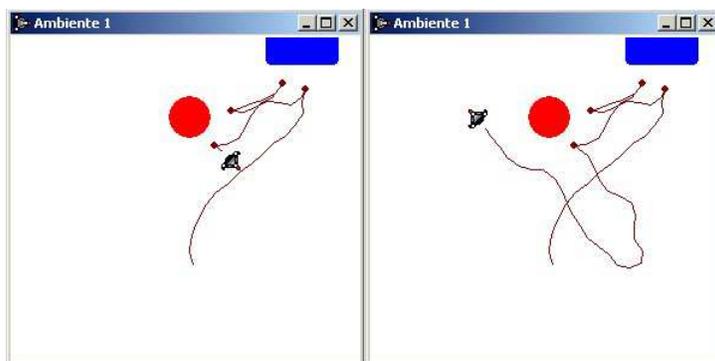


Figura 5.19 – O robô captura o lixo, mas não é estimulado pela base e começa a explorar o ambiente. Nessa trajetória ele se depara novamente com o lixo, mas o ignora, pois já está carregando lixo.

Passando por algumas colisões, o robô não consegue retornar a base para depositar o lixo e/ou recarregar suas baterias dentro do tempo determinado (veja Figura 5.20).

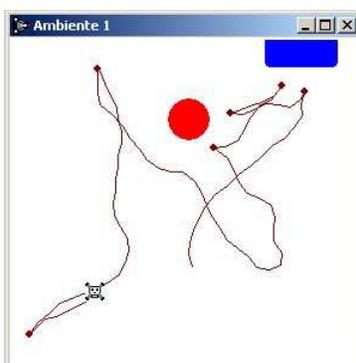


Figura 5.20 – O robô explora o ambiente mas as colisões e a falta de tempo não permitem que ele retorne à base novamente.

Com essa simulação, as duas situações mais conflitantes exigidas para essa tarefa foram verificadas. A primeira é caracterizada pelo fato do robô com energia baixa ignorar o lixo e encontrar a base. A segunda ocorre quando o robô já está carregando lixo e se depara novamente com o lixo. Nesse caso, o lixo também deve ser ignorado.

Vale um comentário em relação ao comportamento errático verificado na parte final da simulação. O robô se encontra carregando lixo e não retorna mais à base simplesmente porque não é capaz de detectar a localização da base a partir da leitura de seus sensores. Repare na Figura 5.20 que, durante toda esta fase final, o robô navega perpendicularmente à base ou na direção oposta a esta, com exceção do último trecho após a colisão.

A rede proposta em WATANABE *et al.* (1999) foi submetida às mesmas condições iniciais. O resultado mostra que a rede também obedeceu à prioridade do robô e buscou reabastecer suas baterias ao invés de capturar o lixo (veja Figura 5.21).

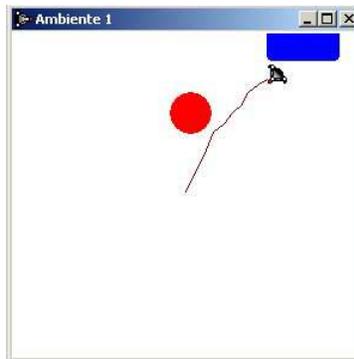


Figura 5.21 – O robô ignora o lixo, buscando a base e recarregando suas baterias tal como a rede evoluída.

Após a recarga, assim como na simulação com a rede evoluída, com energia suficiente o robô direciona-se para a captura do lixo (veja Figura 5.22). Mesmo não indo diretamente para a base, provavelmente por não receber o estímulo necessário, o robô após explorar parte do ambiente encontra o caminho da base.

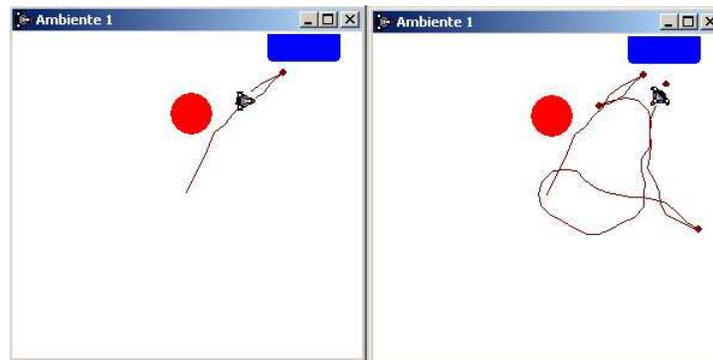


Figura 5.22 – O robô, inicialmente não sendo estimulado o suficiente pela base, explora parte do ambiente e, quando a localiza, efetua o depósito do lixo e a recarga.

Porém, feito o depósito do lixo e a recarga o robô não se direciona para o lixo, e permanece navegando em comportamento exploratório até esgotar o tempo (veja Figura 5.23). Vale aqui o mesmo comentário realizado junto à Figura 5.20.

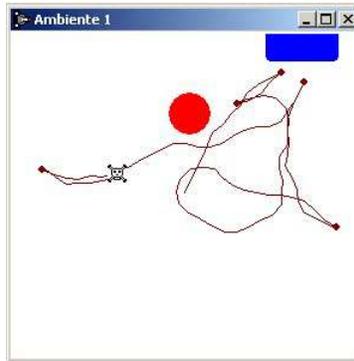


Figura 5.23 – O robô, inicialmente não sendo estimulado o suficiente pelo lixo, permanece navegando em comportamento exploratório até o tempo se esgotar.

Verifica-se então que a rede evoluída realizou uma captura de lixo a mais que a rede proposta. Isso identifica, ainda que de forma não conclusiva, que o processo evolutivo foi capaz de encontrar uma rede que melhor se adequasse às condições do ambiente e às características do modelo computacional do robô.

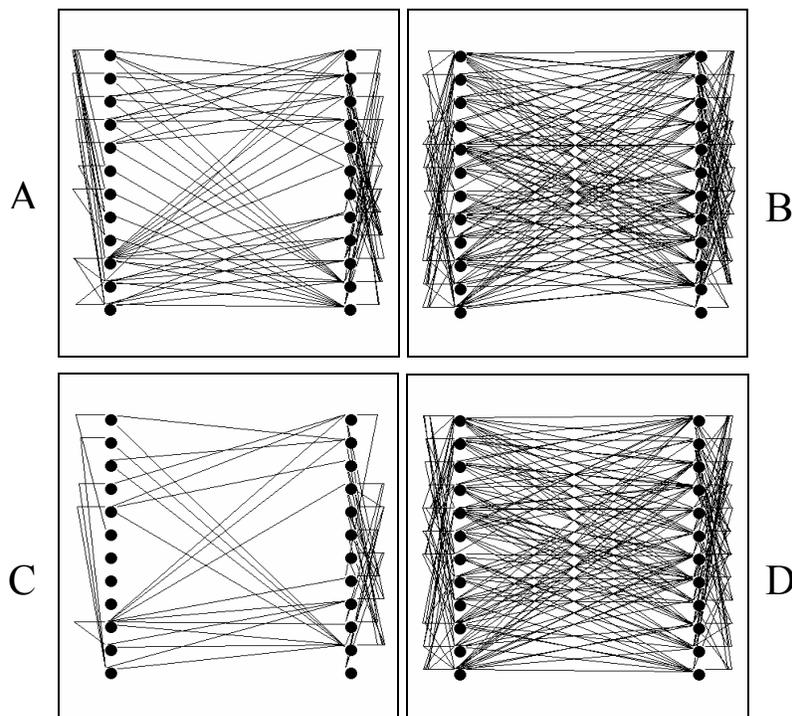


Figura 5.24 – A comparação das estruturas evidencia que há várias possibilidades de configurações para que a rede seja capaz de realizar a tarefa de coleta de lixo. A estrutura A representa a rede proposta em WATANABE *et al.* (1999), com 100 conexões e a B a rede obtida pelo processo evolutivo, com 282 conexões. As duas possuem 47 conexões iguais (estrutura C) e 288 conexões diferentes (estrutura D).

As estruturas das redes, quando comparadas, evidenciam a mesma característica da rede avaliada na Seção 5.1, onde o número de conexões é maior na rede evoluída (veja Figura 5.24). Quando se compara a estrutura da rede evoluída utilizada nessas simulações com a estrutura da rede evoluída utilizada na Seção 5.1, observa-se que elas possuem o mesmo número de conexões iguais (47), embora não as mesmas, quando comparadas com a rede proposta em WATANABE *et al.* (1999).

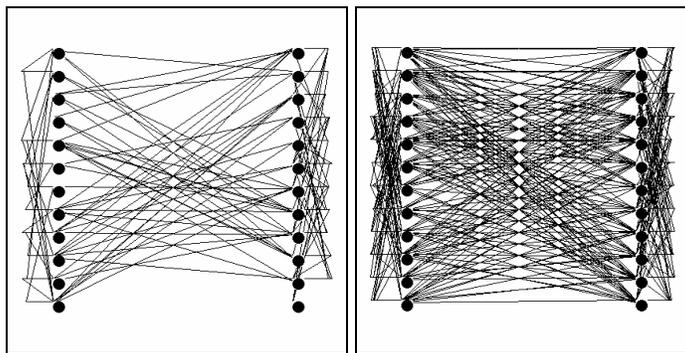


Figura 5.25 – A comparação das estruturas indica que há várias configurações possíveis para uma rede, dada uma mesma tarefa. A estrutura A exibe as 102 conexões iguais entre as redes das Figuras 5.3B e 5.24B. A estrutura B exibe as 309 conexões diferentes.

A Figura 5.25 exibe as estruturas com as 102 conexões iguais entre as duas estruturas obtidas pelo processo evolutivo, que representam aproximadamente 50% do número total de conexões das duas redes. As conexões diferentes são em número de 309, o que indica que há várias configurações possíveis para uma rede, dada uma mesma tarefa.

As comparações numéricas entre as conexões das estruturas das redes são importantes, pois confirmam a variabilidade presente nos resultados alcançados e possibilitam especulações sobre o número mínimo de conexões necessárias para a realização das tarefas, e também sobre a maior ou menor frequência de uma ou outra conexão nas várias propostas de configuração de rede. Um estudo mais aprofundado dessas comparações e constatações faz parte das perceptivas futuras desse trabalho.

5.3 Experimentos Práticos

A seguir, serão apresentados os resultados obtidos nos experimentos preliminares com o robô Khepera II para a tarefa de desvio de obstáculos. Para a realização dos testes, foi construído um ambiente retangular de isopor revestido com papel branco, medindo 40cm de comprimento e 30cm de largura.

O robô tem aproximadamente 7cm de diâmetro e desloca-se a 0,5cm a cada passo. Os sensores são em número de 8, dispostos ao redor do robô de maneira que 6 estão situados à frente e 2 na parte de trás (veja Figura 5.26 e Apêndice B). Para os testes, serão considerados apenas os sensores da parte dianteira. As informações provenientes dos sensores são: distância e direção dos obstáculos. Mais informações sobre os aspectos técnicos do robô Khepera II são encontradas no Apêndice B.

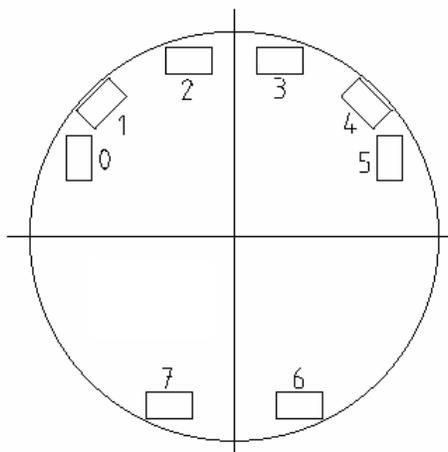


Figura 5.26 – Disposição dos sensores no robô Khepera.

Os primeiros experimentos exibidos têm como objetivo demonstrar a viabilidade da transferência do controle para ambientes reais, a capacidade de coordenação da rede obtida através do processo evolutivo e a aplicabilidade das redes imunológicas a sistemas de controle.

Nesses experimentos, o robô é submetido a situações de navegação variadas a fim de demonstrar como os comportamentos elementares, adequadamente coordenados, fazem surgir comportamentos mais complexos. A rede utilizada para efetuar o controle é a mesma das simulações na Seção 5.2.1, para desvio de obstáculos.

No experimento descrito na Seção 5.3.4, a situação do experimento descrito na Seção 5.3.3 é repetida com o objetivo de realizar o acompanhamento do nível de concentração dos anticorpos. Para cada passo do robô, serão exibidos os antígenos que adentraram o sistema e a nova configuração dos níveis de concentração dos anticorpos, assim como o anticorpo vencedor que atua no ambiente, identificado no gráfico pela cor azul.

5.3.1 Navegar pelo Corredor

No primeiro experimento (veja Figura 5.27), o robô é submetido à tarefa de navegar por um corredor tangenciando as paredes e escapando para a outra área do ambiente. Nesse percurso, ele precisa desviar de obstáculo na saída do corredor. Esse teste tem o objetivo de comprovar a capacidade de coordenação da rede imunológica, fazendo com que o robô navegue em segurança.

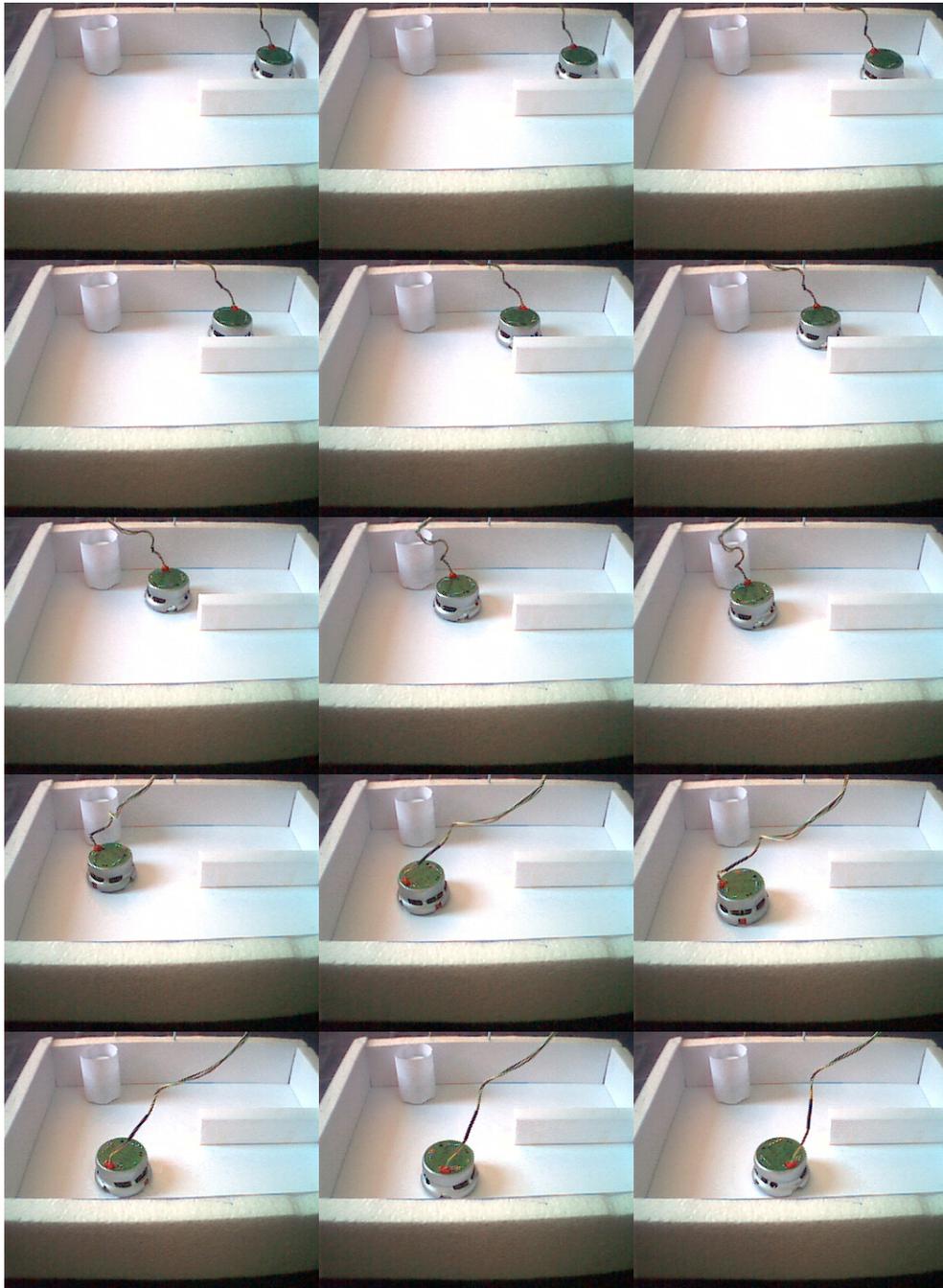


Figura 5.27 – No experimento realizado, o robô foi forçado a tangenciar paredes e desviar do obstáculo posicionado ao final do corredor.

5.3.2 Navegar ao redor do obstáculo

Em outro experimento realizado, o robô tem como objetivo contornar o obstáculo posicionado no centro do ambiente (veja Figura 5.28). A distância entre o obstáculo e a

parede do maior lado do ambiente é cerca de 10cm. O robô completou a trajetória mostrando coerência nas ações.



Figura 5.28 – Rodeando um obstáculo: o robô conseguiu realizar a navegação sem colisões.

5.3.3 Dilema do “beco”

Um terceiro experimento foi realizado buscando verificar a capacidade da rede em teste de sair do dilema do “beco”. Nesta situação, o robô encontra-se cercado, dessa maneira todos os sensores frontais captam obstáculos.

O algoritmo desenvolvido por BRAITENBERG (1984) é disponível no Khepera em seu modo de exibição. Para efeito de comparação, o Khepera executando esse algoritmo foi colocado no ambiente em situação que provoca o dilema descrito. O robô executando o algoritmo de BRAITENBERG (1984) consegue desviar do beco contornando a parede e escapando da armadilha (veja Figura 5.29).

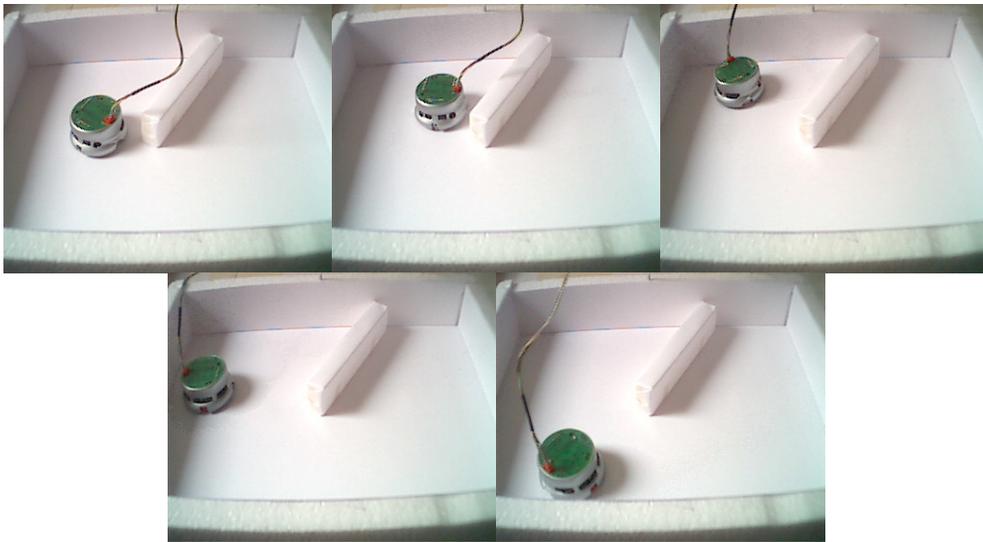


Figura 5.29 – O robô evita o “beco”, contornando a parede e conseguindo escapar.

A rede evoluída em teste também consegue, através de sua dinâmica, a partir da qual as conexões da rede levam a alteração das concentrações, fazer com que o robô evite o beco (veja Figura 5.30), assim como no teste anterior com o algoritmo de BRAITENBERG (1984).



Figura 5.30 – O robô evita o “beco”, contornando a parede e conseguindo escapar.

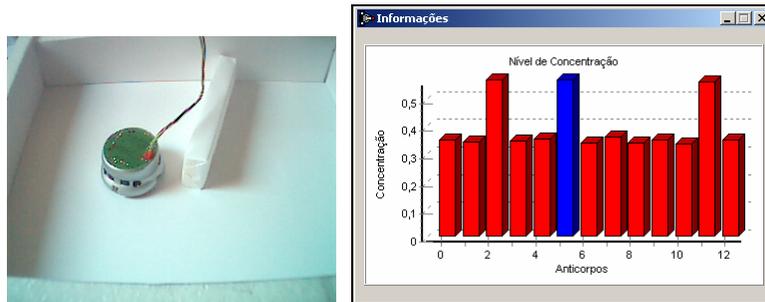
A rede utilizada para os testes é a mesma da Seção 5.2.1, na tarefa de desvio de obstáculo. Dessa maneira, os testes preliminares com o Khepera vêm comprovar a capacidade de coordenação que a rede já havia demonstrado nas simulações, a qualidade do modelo computacional, tanto do ambiente quanto do robô, e a flexibilidade na implantação do sistema de controle obtido em sistemas reais.

5.3.4 Acompanhamento da concentração

Com o intuito de exemplificar a dinâmica da rede imunológica, o experimento do beco da Seção 5.3.3 foi repetido e o acompanhamento do nível das concentrações dos anticorpos foi realizado passo a passo. O conjunto de antígenos que adentram o sistema também é exibido e o anticorpo vencedor, o qual atuará no ambiente, será identificado pela

cor azul no gráfico. Todo gráfico apresentado é responsável pela posição do robô apresentada na figura seguinte. Isso porque a figura ao lado de cada gráfico corresponde à situação corrente em que o robô realiza a leitura dos sensores. As informações entram no sistema, são processadas pela rede imunológica e esta gera a configuração das concentrações apresentadas no gráfico. Devido ao tipo de experimento realizado, nem todos os anticorpos terão níveis expressivos de concentração.

Cada conjunto de figura e gráfico será chamado de passo. O primeiro passo exibe a condição inicial e a concentração dos anticorpos. Este perfil de concentração foi obtido após a leitura das informações dos sensores, quando a rede já realizou seu processamento e o anticorpo que atuará no ambiente já foi escolhido.

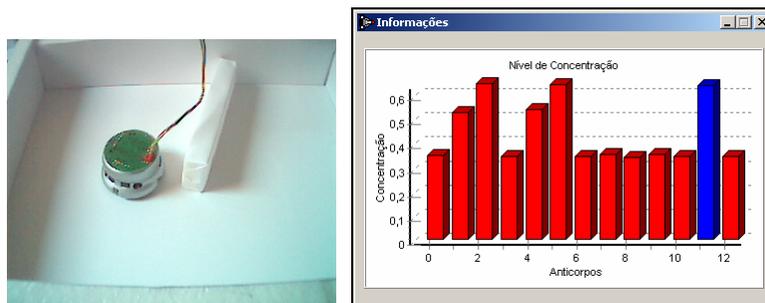


Passo - 1

Dada a entrada dos antígenos, 2 1.0.0 – 0.1, 2 0.0.1 – 1.1 e 2 0.1.0 – 0.1, verifica-se que os anticorpos 3, 6 e 12 são os mais concentrados (Obs: deve-se somar 1 ao índice do eixo de ordenadas do gráfico de barras). As regras associadas são:

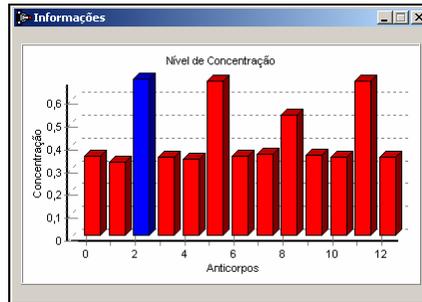
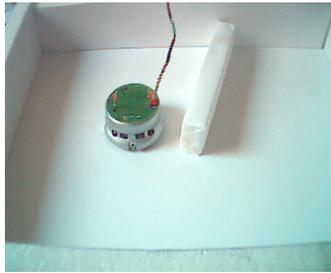
3-⟨Obstáculo à frente, muito longe⟩-⟨Vá em frente⟩	2 010	01 101
6-⟨Obstáculo à frente, muito longe⟩-⟨Vá em frente⟩	2 010	01 101
12-⟨Obstáculo à direita, perto⟩-⟨Vire à esquerda, forte⟩	2 001	11 100

sendo que o anticorpo 6 foi o escolhido para atuar. É interessante observar que o anticorpo que responde ao obstáculo à direita não foi um dos mais concentrados. Porém a ação tomada pelo robô foi coerente. Isso demonstra que o processo evolutivo adequou as conexões de forma a responderem corretamente à situação, embora para um projetista isso possa parecer inadequado. Dessa forma, a hipótese de que a partir de comportamentos elementares possam surgir comportamentos complexos está configurada nessa situação.



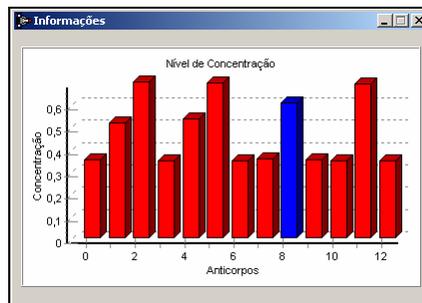
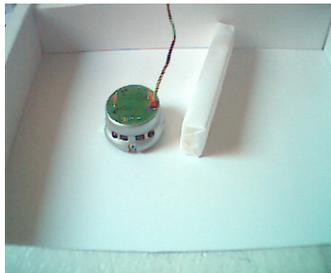
Passo - 2

No passo seguinte, com a entrada dos antígenos 2 1.0.0 – 0.1, 2 0.0.1 – 1.1, 2 0.1.0 – 0.1 e 2 0.1.0 – 1.0, a concentração é ligeiramente alterada para alguns anticorpos, mas os três mais concentrados no passo anterior continuam sendo os anticorpos com maiores concentrações. Contudo, dessa vez o anticorpo 12 foi o escolhido para atuar, o robô então vira poucos graus para a direita.



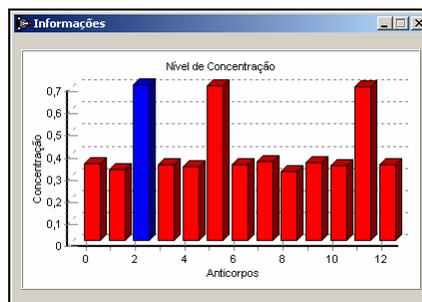
Passo - 3

Agora os antígenos 2 1.0.0 – 0.1, 2 0.0.1 – 1.0 e 2 0.1.0 – 0.1 provocam a alteração de alguns anticorpos, mas nada que possa reduzir a concentração dos anticorpos 3, 6 e 12. Nesse passo, o anticorpo 3 ganha e atua fazendo o robô dar um passo à frente.



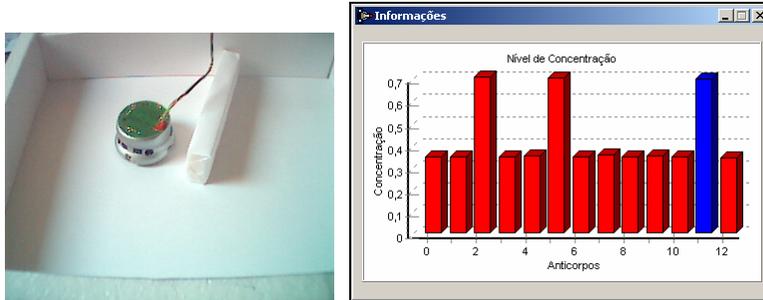
Passo - 4

Como os antígenos permaneceram os mesmos, as concentrações dos anticorpos, mantiveram a tendência de alta e os mais concentrados permaneceram com as maiores concentrações. Contudo, um dos anticorpos com concentração pouco inferior, devido ao método probabilístico (Seção 4.2.5) utilizado na seleção, foi o escolhido. O robô então vira para a direita.



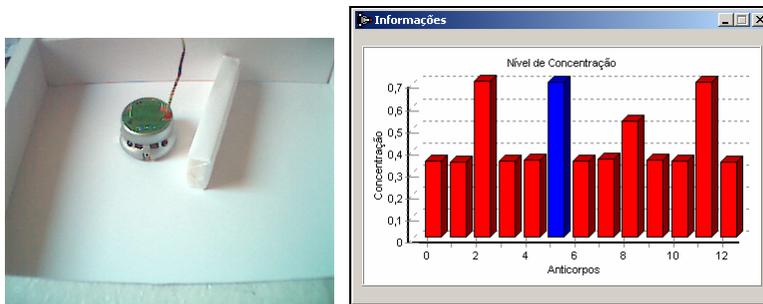
Passo - 5

Diante do retorno do robô à mesma direção da posição inicial, os antígenos voltam a ser, 2 1.0.0 – 0.1, 2 0.0.1 – 1.1 e 2 0.1.0 – 0.1, e novamente os anticorpos 3, 6 e 12 são os mais concentrados. Assim, o método de seleção escolhe o anticorpo 3 para atuar. O robô então dá um passo à frente e mantém-se na mesma direção.



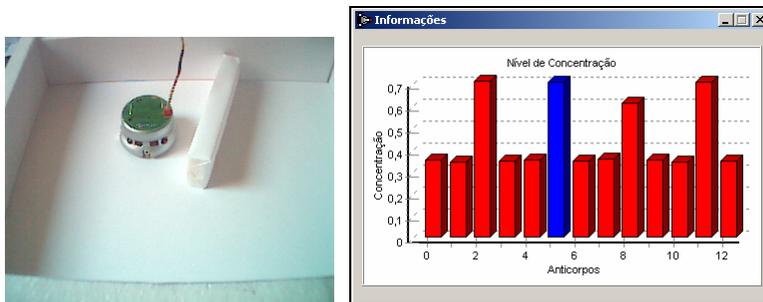
Passo - 6

Como a direção não foi alterada os antígenos captados foram os mesmos, o que resultou na estabilização dos anticorpos 3, 6 e 12 como os mais concentrados. Porém dessa vez o anticorpo 12 foi selecionado e o robô virou para a esquerda.



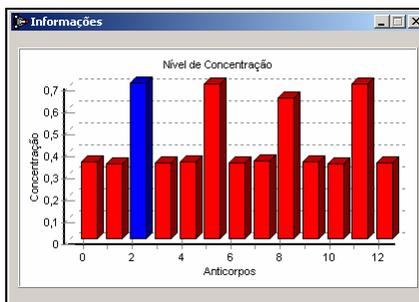
Passo - 7

Os antígenos captados na nova situação foram 2 1.0.0 – 0.1, 2 0.0.1 – 1.0 e 2 0.1.0 – 1.0, que levaram a poucas alterações nas concentrações dos anticorpos. Dessa maneira o anticorpo 6 foi vencedor e levou o robô mais um passo à frente. O robô deu ainda, uma seqüência de dois passos à frente.



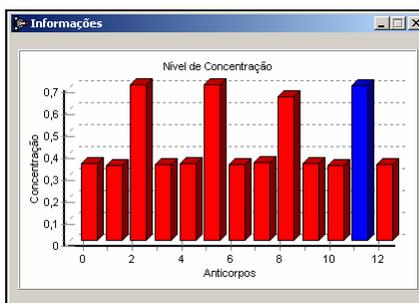
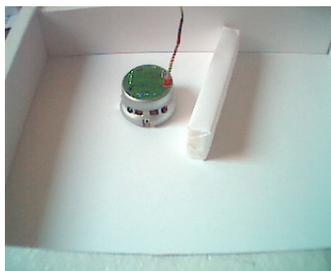
Passo - 8

Como a situação representada pelos antígenos 2 1.0.0 – 0.1, 2 0.0.1 – 1.0 e 2 0.1.0 – 0.1, devido à pequena distância percorrida em cada passo (0,5cm), permanece a mesma, as concentrações mantêm sua tendência de alta, e o robô continua a andar para a frente.



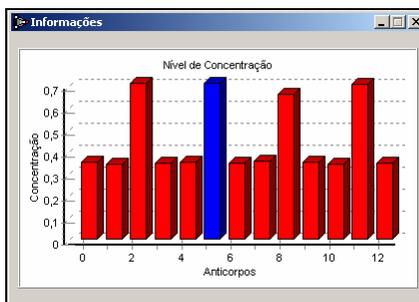
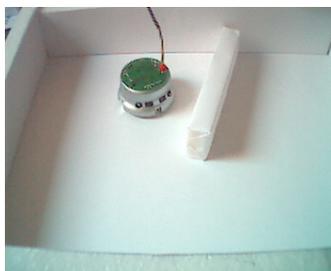
Passo - 9

A concentração do anticorpo 12 sinaliza a necessidade de virar para a esquerda. Se, por acaso, ele não fosse selecionado nessa iteração, quando o robô se aproximasse da parede a dinâmica das conexões faria com que a concentração de um anticorpo com essa ação conseguisse ser uma das maiores, a fim de desviar da parede.



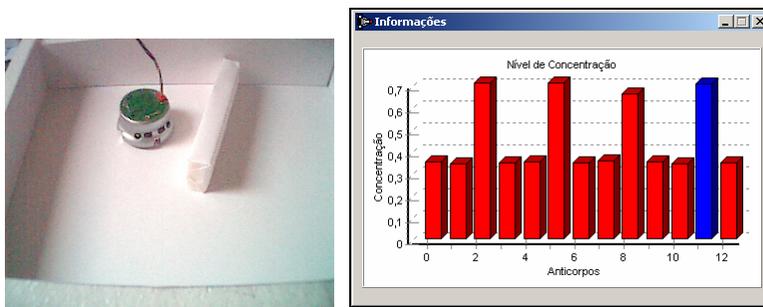
Passo - 10

Como os antígenos 2 1.0.0 – 0.1, 2 0.0.1 – 1.0 e 2 0.1.0 – 0.1, que entraram no sistema, sinalizavam obstáculos longe do robô, a mudança de direção não alterou o panorama da situação captada para o robô e, dessa maneira, as concentrações dos anticorpos se mantiveram.



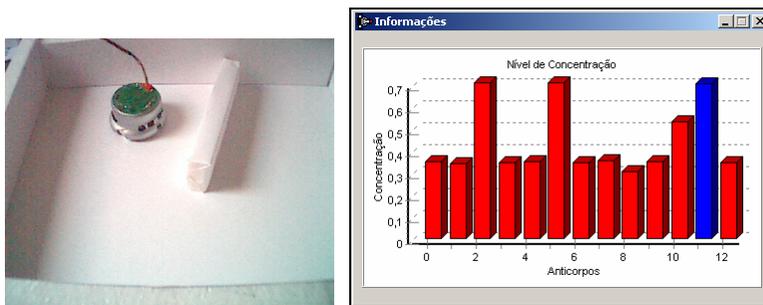
Passo - 11

Apesar da ação de virar à esquerda ser ainda uma das maiores, o anticorpo 6 foi selecionado e o robô deu um passo à frente. Observa-se que a ação é coerente, uma vez que os sensores indicam que os obstáculos estão longe.



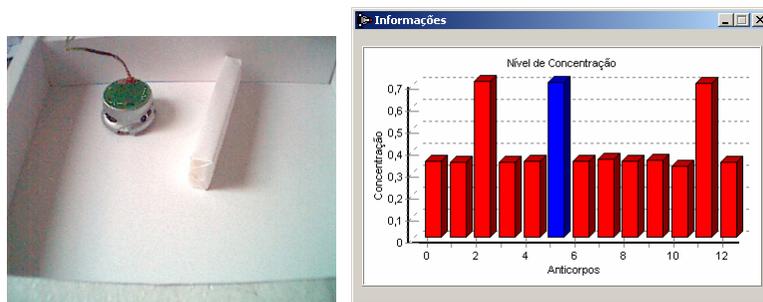
Passo – 12

Mantendo-se a tendência da concentração, o anticorpo 12 que indica a ação de virar à esquerda é selecionado, e o robô começa então a fazer o retorno, escapando do beco.



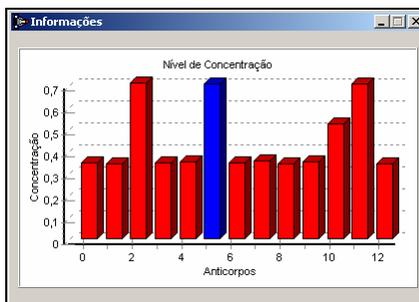
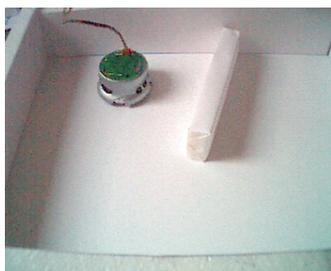
Passo – 13

Nesta situação, a leitura dos sensores do robô indica os antígenos 2 1.0.0 – 1.0, 2 0.0.1 – 1.1 e 2 0.1.0 – 1.0. Isso faz com que as concentrações retomem o padrão do início do experimento.

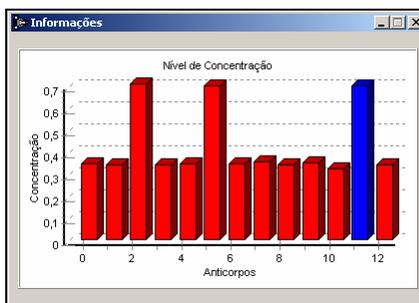
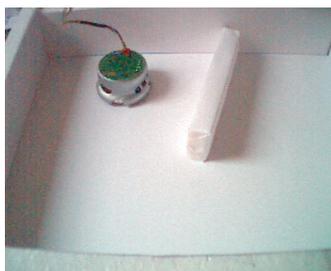


Passo - 14

O método de seleção probabilístico escolhe por duas vezes seguidas o anticorpo 6 para atuar, e com isso o robô dá mais dois passos à frente.

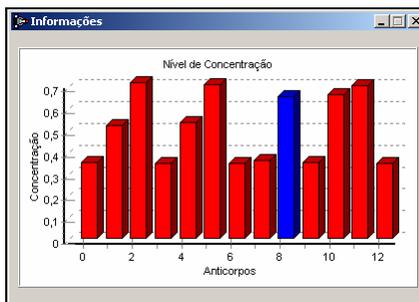
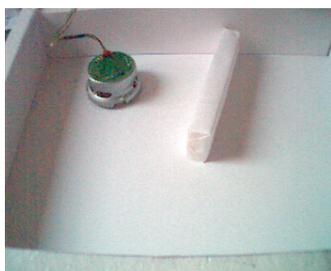


Passo - 15



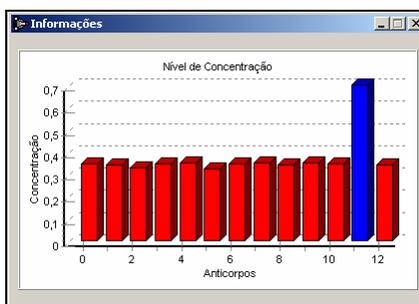
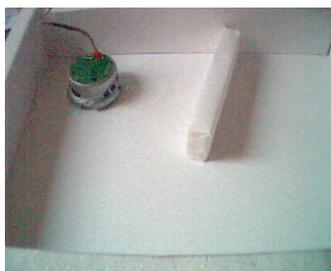
Passo - 16

O nível de concentração dos anticorpos foi mantido nos três últimos passos. Com isso, o anticorpo 12, com bom nível de concentração, foi escolhido pelo método de seleção, e o robô então vira para a esquerda. Com a proximidade da parede, os antígenos 2 1.0.0 – 1.0, 2 0.0.1 – 1.0 e 2 0.1.0 – 1.0 provocam certas alterações nos níveis de concentração.

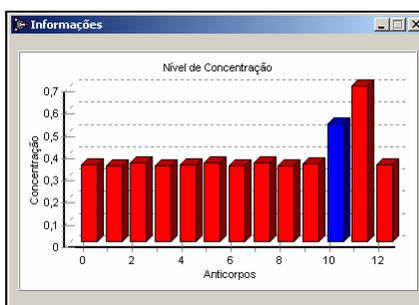


Passo - 17

Essa tendência é mantida até que apenas a concentração do anticorpo 12 mantém-se elevada, forçando o robô a virar para a esquerda. Isso porque a entrada dos antígenos 2 1.0.0 – 0.1, 2 0.0.1 – 1.0 e 2 0.1.0 – 1.0, juntamente com a dinâmica das conexões, força as concentrações dos outros anticorpos para baixo. No passo seguinte, com a entrada do antígeno 2 0.0.1 – 1.1 no lugar de 2 0.0.1 – 1.0, verifica-se que, mantendo-se a situação, o anticorpo 11, que também é responsável por virar à esquerda, inicia o aumento de sua concentração.

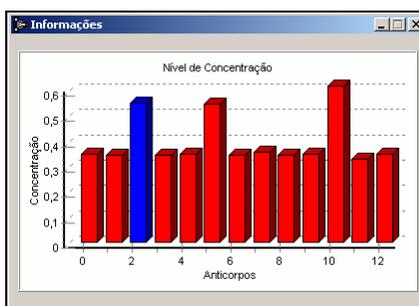


Passo - 18



Passo - 19

Retomando-se uma situação semelhante à condição inicial, percebe-se que as concentrações dos anticorpos 3, 6 e 12 voltam a crescer, de maneira que o anticorpo 3 é selecionado, fazendo com que o robô dê um passo à frente.



Passo - 20

Nesse experimento, verificou-se que a rede imunológica reage adequadamente aos antígenos (situação corrente do robô), resultando na escolha da ação correta. A capacidade da rede imunológica de realizar o controle leva em consideração a propagação, através das concentrações dos anticorpos, do conhecimento das experiências passadas.

Ao analisar os passos exibidos, percebe-se que os três anticorpos mais concentrados, desde a condição inicial, mantiveram suas taxas de concentração praticamente ao longo de todo o experimento, sinalizando as ações mais adequadas, dado que o bico exerce as mesmas excitações sobre o robô por uma longa seqüência de passos. Porém, quando a situação tornou-se crítica, nos passos 18 e 19, apenas o anticorpo responsável por virar à esquerda continuou com a concentração elevada.

5.4 Considerações

Nesse capítulo, procurou-se evidenciar a aplicabilidade da rede imunológica como mecanismo de coordenação, e buscou-se a validação da aplicação do processo evolutivo na obtenção automática de redes imunológicas capazes de atender os objetivos especificados para cada situação.

A análise, ainda que superficial, das conexões encontradas nas redes obtidas através do processo evolutivo reafirmam a concepção de que estruturas modulares que trabalham com processamento distribuído têm sua capacidade reduzida quando sua construção é baseada em uma visão de âmbito local, através de, por exemplo, um método construtivo. Dessa forma, com a aplicação de algoritmos evolutivos, além de romper essa barreira, abriu-se a possibilidade de estudar o aspecto cognitivo da estrutura e indicou-se a existência de soluções alternativas caracterizadas por ações de controle autônomo equivalentes, embora provenientes de topologias de rede imunológica bem distintas.

Capítulo 6

Considerações finais

6.1 Múltiplos objetivos, dinâmica e sistemas híbridos

Ao se buscar caracterizar a identidade deste trabalho, é possível recorrer a uma cadeia de eventos derivados da necessidade de se automatizar a obtenção de um controlador para a tarefa de navegação autônoma de um robô móvel, sendo esta tarefa sujeita ao atendimento de múltiplos objetivos. Esses eventos encadeados são listados a seguir:

1. Todo controlador para navegação autônoma deve ser capaz de atuar sem a necessidade de ter completo domínio sobre os possíveis efeitos que a seqüência de ações terá na definição da seqüência de estados do ambiente.
2. Todo controlador para navegação autônoma deve ser capaz de atuar sem a necessidade de ter completo domínio sobre os possíveis efeitos que a seqüência de ações terá no nível de atendimento dos objetivos da navegação.
3. A existência de múltiplos objetivos de navegação, possivelmente conflitantes, dificulta sobremaneira a síntese do controlador.
4. Logo, a adoção de um controlador fundamentado em mapeamentos estáticos do tipo <condição>-<ação> fica comprometida pela inexistência de uma definição precisa do que seria a <condição> a ser considerada a cada momento em que uma <ação> deve ser adotada, e vice-versa. A <condição> não pode se restringir ao estado atual dos sensores ou a um conjunto predeterminado de valores atuais e passados do estado dos sensores, como normalmente é empregado no caso de mapeamentos estáticos. Não há um procedimento elementar que conduza à modularização da tarefa de navegação. Sendo assim, a determinação do que seria a melhor <ação> a ser adotada nem sempre pode ser realizada com sucesso a partir de uma definição prévia da <condição>, já que nem toda a informação relevante vai estar disponível a cada momento, ou então se está considerando informações irrelevantes na definição do que viria a ser a <condição>.
5. Frente a este cenário, a alternativa é a proposição de um controlador concebido com base em um processo de mapeamento dinâmico, ou seja, em que toda <condição>,

presente e passada, e toda <ação> passada possa influenciar, de algum modo, na determinação da melhor <ação> a ser adotada, num dado momento.

6. Entretanto, a síntese desta dinâmica geralmente não pode ser realizada manualmente pelo projetista e a eficácia de técnicas construtivas de projeto é muito limitada, pois deve-se tomar decisões locais para influir no comportamento global.
7. No contexto deste trabalho, este mapeamento dinâmico foi realizado por uma rede imunológica. A interpretação é a seguinte: caso se adote uma rede imunológica com anticorpos do tipo <condição>-<ação> para realizar este mapeamento dinâmico, ela seguramente vai fazer uso de <condições> e <ações> passadas durante o processo de tomada de decisão, devido à existência de uma dinâmica de interação vinculada à estrutura de conexões da rede.
8. No entanto, este uso de <condições> e <ações> passadas não pode ser arbitrário, ou seja, a dinâmica da rede deve ser tal que conduza ao atendimento dos objetivos de navegação. Isto implica que deve existir um padrão de conexões capaz de maximizar o atendimento dos objetivos durante a operação da rede imunológica como definidora das ações de navegação, embora não existam procedimentos sistemáticos para se determinar este padrão de conexões.
9. Caso se adote a rede imunológica como a responsável pelo mapeamento dinâmico, ou seja, pela definição do anticorpo cuja ação deve ser adotada a cada momento, a última etapa então desta longa cadeia de eventos é a automatização do processo de definição do padrão de conexões da rede imunológica. E a computação evolutiva pode então ser empregada para se realizar uma busca populacional do melhor padrão de conexões dentre todos os padrões de conexões possíveis.
10. Resulta então um sistema híbrido que tem a rede imunológica como responsável por introduzir dinâmica junto ao processo de tomada de decisão e tem agora a computação evolutiva como responsável por definir a dinâmica que conduza ao atendimento dos objetivos de navegação. Em outras palavras, a existência de múltiplos objetivos levou à necessidade de um mapeamento dinâmico; a necessidade de um mapeamento dinâmico levou à proposição de redes imunológicas em sua implementação; e a necessidade de atender aos múltiplos objetivos a partir da atuação da rede imunológica levou à busca evolutiva do melhor padrão de conexões.

O sistema híbrido resultante é então capaz de reunir as potencialidades mais expressivas da rede imunológica e da computação evolutiva. Da rede imunológica, foram incorporadas a capacidade de sintetizar sistemas dinâmicos de elevada complexidade, a distribuição da informação e a capacidade de adaptação gradual às excitações externas recebidas. Da computação evolutiva, foi incorporada a possibilidade de realizar buscas eficazes por soluções de boa qualidade em espaços discretos, de elevada dimensão e guiadas apenas por um índice numérico de qualidade, que é atribuído a cada solução-candidata (padrão de conexão da rede imunológica). Por se tratar de uma abordagem populacional, existe uma população de soluções-candidatas a cada geração do processo evolutivo.

Todas essas propriedades reunidas e aplicadas de forma adequada possibilitaram a criação de um sistema capaz de se adequar a ambientes dinâmicos e encontrar a melhor estrutura que garanta eficiência e flexibilidade na execução de tarefas de navegação autônoma.

6.2 Principais contribuições

A principal contribuição deste trabalho de pesquisa está na proposta de evolução de redes imunológicas para o controle autônomo de robôs móveis. A aplicação de redes imunológicas a problemas de navegação autônoma já havia sido proposta por WATANABE *et al.* (1999). No entanto, a abordagem aqui apresentada difere daquela de WATANABE *et al.* (1999) à medida que se buscou definir automaticamente as conexões entre os anticorpos da rede imunológica, os quais são responsáveis pelos comportamentos elementares do processo de navegação, via técnicas de computação evolutiva, ao contrário da anterior que utiliza técnicas construtivas de aprendizado por reforço. As redes evoluídas, dadas as características de busca em paralelo dos algoritmos evolutivos, levam em consideração aspectos globais enquanto a proposta construtiva de aprendizado por reforço conserva o aspecto local de projeto.

A proposta desta dissertação também difere daquela apresentada por WATANABE *et al.* (1998), pois lá empregou-se uma perspectiva distal para definição dos módulos de comportamento elementar. Com isso, cada anticorpo estaria associado a uma decisão de alto nível, a qual requer múltiplas atitudes do robô para ser executada. A vantagem da perspectiva distal está no fato do algoritmo evolutivo poder ser aplicado tanto na síntese das conexões como dos nós da rede, pois ambos são em número reduzido. Por adotar, nesta dissertação, um híbrido entre as perspectivas distal e proximal, aumenta-se significativamente o número de nós e conexões, embora facilite a definição a priori dos nós da rede. Com isso, os algoritmos evolutivos se ocuparam apenas em buscar o melhor padrão de conexão para cada rede imunológica, dados os nós da rede.

Os mecanismos de síntese e operação do controlador imuno-genético proposto estão de acordo com os paradigmas da robótica baseada em comportamento, por estarem vinculados à necessidade de propor um módulo de coordenação de comportamentos elementares de navegação.

Os resultados das simulações computacionais apresentados evidenciaram a capacidade de coordenação da rede imunológica resultante da aplicação do processo evolutivo. Apesar do fato de que as redes imunológicas resultantes apresentam estruturas com alta densidade de conexões, as redes apresentaram desempenho estável nos testes realizados. A presença de alta densidade de conexões é também um indicativo, embora não-conclusivo, de que a definição de um padrão de conexões que leve a rede imunológica a se comportar de forma consistente com os objetivos da navegação não requer a atuação de um projetista e difere significativamente da proposta concebida por especialistas.

Quando transferido para o ambiente real, o sistema apresentou robustez, ainda que em experimentos preliminares.

6.3 Perspectivas futuras

Dentre as perspectivas imediatas, está a realização de experimentos mais avançados com o robô Khepera II, incluindo a tarefa de coleta de lixo. Com isso, haverá condições de se realizar uma análise mais detalhada do impacto da transferência do ambiente de simulação para o ambiente real.

Não está descartada a realização da etapa evolutiva também em um ambiente real, visto que esta etapa foi executada apenas em ambientes de simulação computacional.

Comparações de desempenho e interação com grupos de pesquisa que desenvolvem projetos alternativos de controle autônomo sempre representam etapas fundamentais na consolidação de qualquer nova proposta de controle para navegação autônoma.

O estudo de sistemas com múltiplos robôs é um passo importante na pesquisa de sistemas de navegação autônoma. A emergência de comportamentos complexos a partir da interação de robôs capazes de expressar apenas comportamentos elementares será alvo de investigação, visando inclusive a incorporação de estratégias competitivas e cooperativas. Adaptações junto ao ambiente de simulação computacional serão necessárias para permitir a realização de experimentos considerando múltiplos robôs. Já no ambiente real, nenhuma adaptação ou extensão de técnicas ou componentes se faz necessária com a passagem para a operação com múltiplos robôs.

Além de todas essas possibilidades, existem ainda vários estudos que podem ser realizados visando ajustar outros parâmetros do sistema imuno-genético, incluindo a força de estimulação e supressão entre os anticorpos da rede e a estimulação estabelecida pelo antígeno, os quais são tomados como parâmetros fixos na implementação apresentada neste trabalho.

Outra análise importante deve ser relacionada ao número de conexões das redes obtidas através do processo evolutivo. Embora de forma superficial, as análises sobre duas redes obtidas em processos evolutivos distintos, no Capítulo 5, indicam que pode haver um padrão mínimo de conexões necessárias para o alcance dos objetivos do sistema. Esse tipo de análise pode levar a modificações nas estruturas e nas especificações do processo evolutivo utilizado, no que diz respeito à função de fitness, número de anticorpos e de conexões.

A automatização completa da fase de concepção de redes imunológicas também será investigada. A definição do número e tipos de anticorpos, possivelmente também através de técnicas de computação evolutiva, vai permitir ao sistema maior flexibilidade e portabilidade junto a diversos problemas de navegação autônoma. Não está descartada a possibilidade de se recorrer a outras técnicas de inteligência computacional em conjunto com o sistema imuno-genético aqui apresentado, pois o emprego isolado de algoritmos evolutivos tradicionais, como realizado em WATANABE *et al.* (1998), não é capaz de tratar adequadamente redes imunológicas com o número de nós e conexões requerido aqui.

A hibridização com outros sistemas de controle como, por exemplo, os sistemas classificadores (LCS – *Learning Classifier Systems*) (HOLLAND, 1992) são possibilidades

que já vêm sendo experimentadas, produzindo resultados consistentes e aumentando a flexibilidade do sistema. Alguns resultados preliminares para a navegação autônoma de robôs foram relatados em VARGAS *et al.* (2003), onde uma rede imunológica e um sistema classificador foram tratados conjuntamente, formando um sistema imuno-classificador denominado CLARINET (CLAssifieR Immune NETwork).

A perspectiva mais ambiciosa talvez seja uma de cunho teórico. A síntese de controladores para navegação autônoma com base na operação de uma rede imunológica representa uma das iniciativas mais expressivas no sentido de explorar os aspectos cognitivos do sistema imunológico (DE CASTRO & TIMMIS, 2002). Sendo assim, experimentos de navegação autônoma podem acabar contribuindo na validação de conceitos e teorias da imunologia voltados para a cognição.

Apêndice A

Aspectos técnicos do ambiente de simulação

A.1 Introdução

A simulação computacional é parte importante na concepção de sistemas de controle, incluindo sistemas de controle para navegação autônoma de robôs móveis. Dadas avaliações de custo e tempo de implantação de um ambiente real de navegação, a simulação pode representar um caminho mais rápido, econômico e eficaz na obtenção e avaliação de alguns resultados experimentais. No entanto, é sabido que ambientes de simulação computacional devem necessariamente realizar simplificações em relação aos fenômenos reais envolvidos, não permitindo retratá-los fielmente. Isto implica que resultados de navegação em ambientes virtuais podem não se reproduzir, nem mesmo em seus aspectos fundamentais, quando se emprega o mesmo controlador para comandar um robô real.

Muitos pesquisadores têm como resultado de seus trabalhos a construção de ambientes de simulação que se tornam referências para a área, transformando-se em pacotes de software (*toolboxes*) bastante completos em termos de recursos disponibilizados ao usuário.

Para a realização da pesquisa desenvolvida nesta dissertação, foi concebido um ambiente de simulação computacional adequado para navegação de robôs. Sem intenção de produzir um *toolbox*, muitas características do ambiente são próprias do problema abordado. Contudo, o modo como o ambiente foi implementado possibilita sua utilização para teste de outros sistemas de controle. A maior restrição para este uso está na exigência de se ter que atuar em nível de programação, não podendo ser feito, ao menos por enquanto, o carregamento do módulo de controle no nível de usuário.

A seguir, são descritas as características do ambiente, visando fornecer suporte técnico à sua utilização para teste de outros sistemas de controle. Também são apresentadas as especificações do modelo computacional do robô.

A.2 Apresentação do Ambiente

O ambiente desenvolvido simula a navegação de robôs, fornecendo suporte às mais variadas condições de navegação. Não existem restrições para número de obstáculos e alvos. Uma restrição é o tamanho do ambiente de navegação, onde o limite está associado ao tamanho da janela do software em estado maximizado.

Os menus e botões da área de controle do software são bastante intuitivos e mesmo pessoas que se deparam pela primeira vez com o software não sentem muitas dificuldades. Contudo, não foram programadas rotinas de verificação e validação das operações, o que pode resultar em uma utilização equivocada da interface.

A.3 Linguagem de Programação e Estrutura do Código Fonte

O ambiente foi todo desenvolvido na linguagem Delphi, na versão 5.0. Nenhum componente avançado da linguagem foi utilizado, sendo possível sua alteração e compilação em qualquer máquina que possuir a versão básica instalada.

A linguagem se mostrou bastante flexível fornecendo todos os recursos necessários à implementação. Outro ponto favorável é a facilidade de compreensão do código, devido à elevada estruturação da linguagem.

A programação baseada em orientação a objeto possibilitou a criação de uma classe do tipo *thread*, chamada **Robo**. Esta classe contém toda a informação sobre o robô, facilitando assim a alteração das características que forem necessárias. Dentre essas informações, estão propriedades como número de sensores, velocidade, direção, energia, entre outras, além dos métodos de sensoriamento, navegação e atualização.

Os métodos, assim como os procedimentos de controle, ambiente e menu principal, estão divididos cada um em uma *unit* ou arquivo “.pas” facilitando as alterações e modularizando a implementação. Dessa maneira, ao inserir um novo sistema de controle, basta criar uma nova *unit*, implementar o código e fazer a referência adequada na *unit* **Robo**.

Além disso, várias ferramentas para a aplicação de algoritmos genéticos já estão implementadas e podem ser executadas a partir da passagem dos parâmetros adequados.

A.4 Modelo Computacional do Ambiente

No ambiente, o robô navega sobre o que no Delphi é chamado de *canvas* de um formulário. Com isso, os obstáculos e alvos são estruturas desenhadas sobre o *canvas*. Esse *canvas* ou ambiente pode ter seu tamanho alterado, tendo como limite máximo o tamanho da janela em modo maximizado. Um recurso interessante é a possibilidade de executar duas simulações ao mesmo tempo em janelas diferentes, lado a lado. Esse recurso exige uma máquina com boa capacidade de processamento. Todos os tipos de obstáculos permitem alterações em suas medidas.

Os obstáculos podem ser circulares ou elípticos, retangulares ou quadrados. Um recurso relevante desse ambiente é a capacidade de agrupamento dos obstáculos de maneira a formar novas estruturas.

Os alvos são diferenciados por suas cores e podem ser de dois tipos, azul ou vermelho. Os azuis podem ser quadrados de cantos arredondados ou círculos, sendo que os quadrados permitem a alteração de suas medidas e os círculos não. Os vermelhos são apenas círculos (veja Figura A.1). Nenhum dos dois tipos possui limites de número de exemplares em uma simulação. Todas essas características do ambiente, assim como as do robô, que serão descritas a seguir, são decisões de projeto.

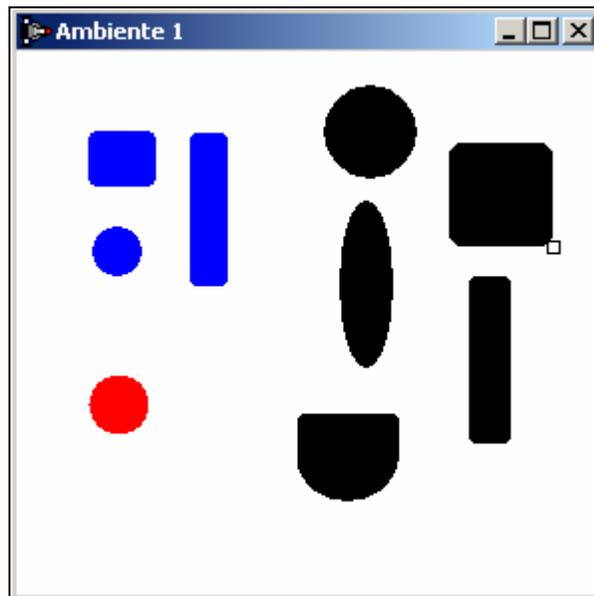


Figura A.1 – Obstáculos (objetos em preto) e alvos (objetos em azul e vermelho).

A.5 Modelo Computacional do Robô

O modelo computacional do robô segue a linha de identificação com o robô real Khepera II. As características do modelo fazem menção às possibilidades encontradas no robô real, a fim de reduzir o investimento na etapa de transição da simulação para o experimento prático.

O modelo de robô não emprega um sistema dinâmico para descrever a trajetória a cada passo, procurando se concentrar mais nas características dos sensores. A trajetória de um robô real certamente acaba sendo influenciada pelas dimensões do robô e pelo posicionamento das rodas, dentre outros aspectos. Na implementação deste trabalho, o robô apresenta uma dimensão, mas todas as operações se referem ao seu centro de massa.

O robô conta com 19 sensores, sendo 13 na parte dianteira e 3 em cada uma das laterais. Esse número pode ser alterado através da propriedade `NSensores` do robô. Os sensores da dianteira estão dispostos a cada 1 grau e os sensores das laterais a cada 10 graus (veja Figura A.2).

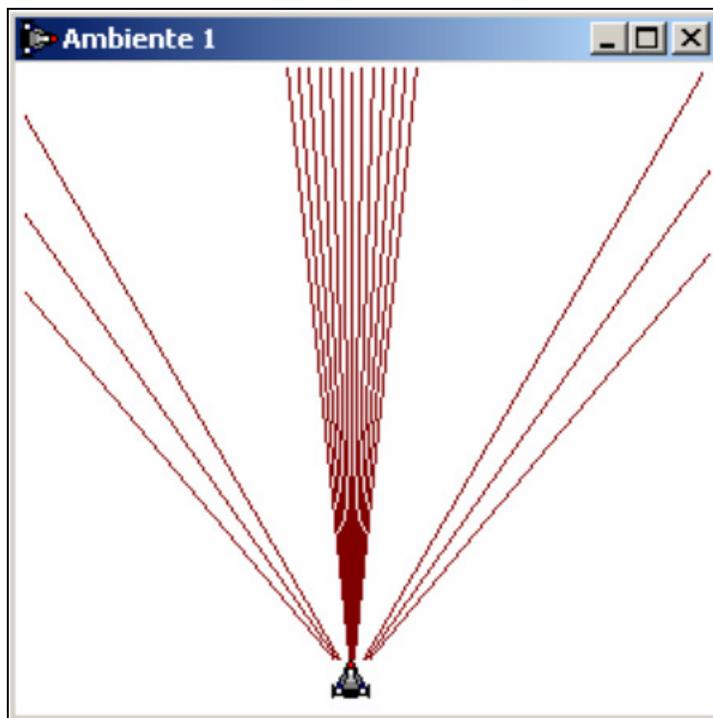


Figura A.2 – Disposição dos sensores do modelo computacional do robô.

Os sensores do modelo são amostras de pontos das zonas de captura dos sensores do robô real. Assim, retas em ângulos predefinidos ($-45,+45$) são traçadas a partir do centro do robô até a margem do ambiente ou até um objeto. Quando um objeto é encontrado, a cor do pixel da tela é analisada e enviada ao robô. Juntamente com as cores, as distâncias desses objetos compõem as informações provenientes dos sensores (veja Figura A.3). Essa característica de captação de cores pode ser implementada no robô real através do acoplamento de um câmera de vídeo e processamento de imagem. As propriedades `Robo.Sensores[ID].distancia` e `Robo.Sensores[ID].cor` contêm a informação abaixo.

Sensor 1 – Cor: Azul Distância: 100

Figura A.3 – Exemplo de estrutura das informações provenientes de um sensor.

Esse procedimento é repetido para todos os sensores no método de sensoriamento. Após essa etapa, um outro procedimento dentro do sensoriamento faz o processamento das informações, verificando qual sensor de uma dada região (central ou lateral) captou a menor distância. Desse modo, a cor desse sensor indicará ao robô qual objeto foi captado na região daquele sensor (frente, esquerda ou direita).

A colisão é uma informação que pode depender do projeto do sistema de controle. Nas simulações desta dissertação, as distâncias menores que 20 pixels determinam a colisão, e pode ser verificada na propriedade `Robo.col`. Se o valor dessa propriedade, de valor inteiro, for 1, então houve a colisão. Caso contrário, será 0.

O deslocamento do robô é dado a partir da ação enviada ao método de navegação. De qualquer maneira, a não ser em caso de colisão, o robô irá deslocar-se para a frente, e alterará ou não a sua direção. O valor desse deslocamento é proporcional à velocidade preestabelecida, em direção à nova posição que o robô deve assumir. Caso um sinal de colisão seja enviado, o valor da rotação do robô é de 180 graus. Todos esses parâmetros são facilmente alterados na *unit* “Navegacao.pas”.

A.6 Execução

O menu “Arquivo” está inteiramente dedicado ao problema estudado. Porém, toda operação com arquivos pode ser facilmente adaptada, visto que essas operações encontram-se em métodos da classe do formulário principal. Os outros menus e os botões da barra são auto-explicativos, sendo acompanhados de dicas quando acionados pelo mouse.

Iniciando a simulação

O primeiro método chamado ao se iniciar uma simulação é o de inicialização, onde são atribuídos os parâmetros que farão parte da simulação, como velocidade, tempo ou número de passos da simulação, energia do robô, entre outros. Por isso, o projetista que desejar alterar esses parâmetros deve recorrer à *unit* “Inicializa.pas”.

A seqüência de execução de um passo do robô em uma dada simulação é iniciada com a leitura dos sensores. A próxima etapa compete ao sistema de controle, que fará o processamento das informações e enviará então ao método de navegação a ação a ser tomada. Depois de calculada a nova posição do robô, pelo método de navegação, o método de atualização fará o reposicionamento do robô na tela e redesenhará a direção dos sensores e a trajetória, quando estes estiverem habilitados.

Após a condição de parada (tempo ou energia) ser alcançada, a *thread* que está executando o robô é eliminada. Assim, para uma nova simulação é necessário adicionar um novo robô e excluir ou não o robô já presente no ambiente (caso não seja excluído, ele será avaliado como obstáculo). Todos os passos descritos acima são executados dentro do método *execute* da *thread* do robô.

A.7 Considerações finais

O modelo do ambiente foi validado através de execuções de navegação empregando diversos tipos de controladores, e mostrou-se eficaz no auxílio ao desenvolvimento e validação de sistemas de controle autônomos para navegação de robôs.

Apêndice B

Aspectos técnicos do robô Khepera

B.1 Apresentação do Khepera

As pesquisas com robôs enfrentavam, há alguns anos, limitações de tecnologia que impossibilitavam os experimentos práticos para a avaliação dos sistemas de controle desenvolvidos. Outra importante limitação era a especialidade em eletrônica que o pesquisador da área deveria apresentar para poder operar os recursos disponíveis.

Há praticamente 10 anos um grupo de pesquisa do *Microprocessor and Interface Laboratory* (LAMI) da Suíça, apoiado pelo *Swiss Research Priority Program*, começou a desenvolver um modelo de robô que poderia ser operado por qualquer profissional com um mínimo de experiência em computação, e que serviria de parâmetro para a comparação dos resultados obtidos pelas pesquisas em navegação, processamento de sinais e análise comportamental, desenvolvidas por vários grupos de pesquisa.

Com o projeto pronto, a empresa K-Team começou a fabricar o Khepera que já está em sua segunda versão (Khepera II) e vem aumentando sua linha de mini e micro robôs a fim de viabilizar os experimentos com robótica e sistemas inteligentes.

A fabricação do Khepera possibilitou a ocorrência de progressos significativos na área de Robótica Evolutiva, propiciando comparações de desempenho no posicionamento das pesquisas em navegação autônoma.

O Khepera é um mini-robô dotado de vários recursos que são suficientes para o desenvolvimento de diversos tipos de sistemas de controle para navegação.

A seguir, serão descritas as principais características, como tipos de sensores e especificações técnicas, e, ao final, alguns detalhes da adaptação do ambiente de simulação desenvolvido para a realização dos experimentos práticos.

B.2 Características

Por ser um mini robô, o Khepera tem seu tamanho reduzido, medindo apenas 70mm de diâmetro e 30mm de altura, e pesando aproximadamente 80g (veja Figura B.1). A opção por dimensões desta magnitude permite a realização dos experimentos mesmo em locais com pouco espaço. Pode ser alimentado por cabo ou por suas baterias internas, as quais possuem autonomia de aproximadamente 1 hora, dependendo dos acessórios acoplados.



Figura B.1 – Robô Khepera II comparado com o tamanho de uma moeda de 2 euros.

Na plataforma básica, ele possui sensores de distância IR e sensores de luminosidade ambiente (veja Figura B.2). Alguns fatores externos, como a presença de lâmpadas incandescentes, influenciam na leitura das informações. Esta influência se deve a características técnicas do hardware utilizado. Tais características podem ser obtidas a partir do manual do Khepera (<http://www.k-team.com/download/khepera.html>).

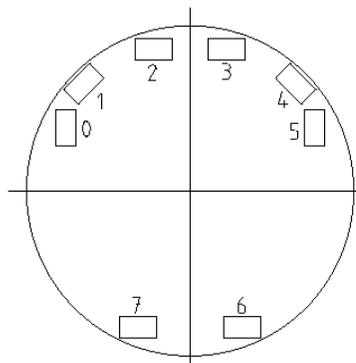


Figura B.2 – Distribuição dos sensores. Seis deles estão dispostos na parte da frente e dois na parte de trás.

O robô é sustentado por duas rodas que realizam a locomoção e dois pontos de contato deslizantes que proporcionam equilíbrio (veja Figura B.3). As rodas giram de

maneira independente e, dessa forma, as curvas realizadas pelo robô são um resultado de diferentes velocidades em cada uma das rodas.

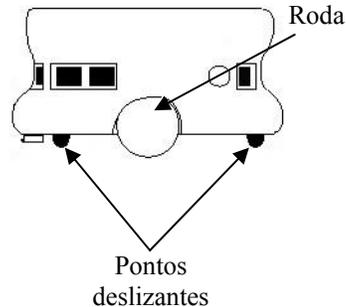


Figura B.3 – O robô se locomove através da tração nas duas rodas. Os pontos deslizantes lhe conferem o equilíbrio.

A velocidade máxima alcançada pelo robô é de 60cm/s e a mínima é de 2cm/s. O modelo dinâmico do robô e outras especificações podem ser obtidas no manual do usuário, disponível na página da empresa (<http://www.k-team.com/robots/khepera/base.html>).

B.3 Aspectos de comunicação e controle

Devido à característica modular do ambiente, descrita no Apêndice A, Seção A.2, para a implantação do sistema de controle desenvolvido no robô real, a maior mudança no software foi a introdução de novos métodos de sensoriamento.

O robô se comunica com o computador através do protocolo de comunicação serial RS-232. A partir deste protocolo, é possível fazer a leitura dos sensores e enviar comandos de atuação para o robô. Essa comunicação pode ser feita via cabo ou através de comunicação via rádio, bastante indicada para experimentos com comportamento coletivo, uma vez que livra o robô do “cordão umbilical”. Não há diferença de protocolo entre os dois métodos de comunicação, porém as configurações dos tempos de resposta por parte do robô requerem alguns cuidados.

Os novos métodos de sensoriamento e navegação incorporaram uma classe de componentes dedicados à comunicação com a porta serial. Dessa forma, a maneira como é conduzida a execução, para cada passo do robô, não foi alterada. As informações provenientes dos sensores do robô são acondicionadas na instância em execução da classe **Robo**, assim como no sensoriamento simulado. De posse dessas informações, o sistema de controle escolhe a ação a ser tomada e envia para o novo módulo de navegação, que foi adequadamente calibrado para reproduzir no robô as ações desejadas.

Validação do controle

Antes de iniciar os experimentos práticos com o sistema de controle autônomo, foi desenvolvido e implementado um software de teste a fim de validar as informações e o controle sobre os atuadores do robô (veja Figura B.4). A partir deste software, foi possível identificar questões como os tempos de resposta aos comandos. Com isso, deslocamento e rotação foram empiricamente calibrados.

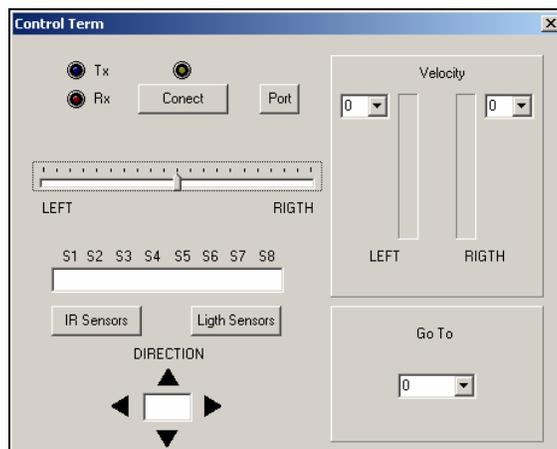


Figura B.4 – Software implementado para a validação do ambiente e da comunicação com o robô.

Deslocamento e rotação

O valor do deslocamento ou passo do robô foi calibrado através do método de tentativa e erro, fixando a velocidade e variando o tempo de ocorrência. Assim, foram encontrados valores de velocidade e tempo que propiciaram ao robô um passo de 0,5 cm.

Do mesmo modo, para calibrar os valores de rotação foi utilizado um gabarito. Em uma folha de papel sobre a qual o robô foi colocado, foram desenhados os ângulos de $\langle 90 \rangle$, $\langle 45 \rangle$, $\langle 22,5 \rangle$, $\langle 11,25 \rangle$ e $\langle 0 \rangle$ grau. Fixando a velocidade para cada uma das rodas, de modo a fazer com que o robô executasse a rotação, e variando o tempo desse movimento, foram encontrados valores de atuação que geravam ações nos ângulos desejados.

B.4 Considerações Finais

Com esses experimentos preliminares, foi possível constatar que o Khepera pode vir realmente a ser um aliado na pesquisa de navegação autônoma e sistemas inteligentes. Isto porque a plataforma é muito flexível, deixando o pesquisador muito à vontade e proporcionando os recursos necessários.

O robô utilizado conta apenas com o acessório de comunicação via rádio. A falta de acessórios como a câmera e a garra (veja Figura B.5) impossibilitam, por exemplo, experimentos como o da coleta de lixo.

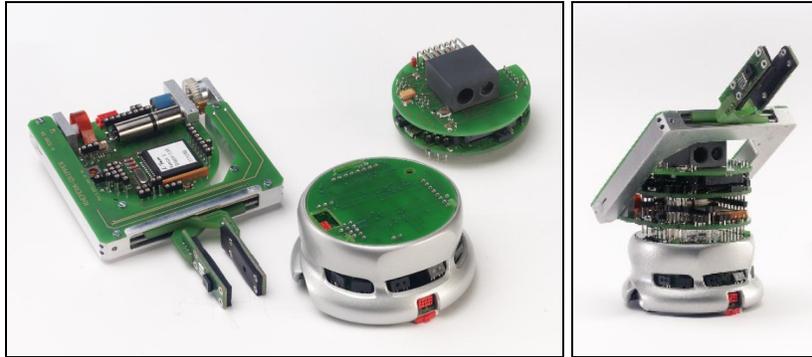


Figura B.5 – Acessórios do Khepera: câmera de vídeo e garra.

Referências bibliográficas

- [1] ACKLEY, D.H., LITTMAN, M.L., “Interactions between learning and evolution”, in C.G. Langton, C. Taylor, J.D. Farmer, S. Rasmussen (Eds.) *Artificial Life II*, SFI Studies in the Sciences of Complexity, vol. X, pp. 487-507, Addison-Wesley, 1992.
- [2] ARKIN, R.C., “Behavior-Based Robotics”, The MIT Press, 1998.
- [3] ATMAR, W., “Notes on the Simulation of Evolution”, *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 130-148, 1994.
- [4] BÄCK T., FOGEL D.B., MICHALEWICZ Z., (Eds), “Evolutionary Computation 1- Basic Algorithms and Operators”, IoP Publishing, 2000a.
- [5] BÄCK T., FOGEL D.B., MICHALEWICZ Z., (Eds), “Evolutionary Computation 2- Advanced Algorithms and Operators”, IoP Publishing, 2000b.
- [6] BARTO, A.G., SUTTON, R.S., ANDERSON, C.W., “Neuronlike elements that can solve difficult learning control problems”, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 13, nº 5, pp. 835-846, 1983.
- [7] BERSINI, H., “Self-Assertion versus Self-Recognition: A Tribute to Francisco Varela”, *Proc. of ICARIS 2002*, pp. 107-112, 2002.
- [8] BRADY, M., “Artificial Intelligence and Robotics”, *Artificial Intelligence and Robotics*, vol. 26, pp. 79-121, 1985.
- [9] BRAITENBERG, V., “Vehicles”, The MIT Press, 1984.
- [10] BROOKS, R., “A robust layered control system for a mobile robot”, *IEEE Transactions on Robotics and Automation*, vol. 2, pp. 14-23, 1986.
- [11] BROOKS, R., “Planning is Just a Way of Avoiding Figuring Out What to Do Next”, Working Paper 303, MIT AI Laboratory, 1987.

- [12] BROOKS, R., “Intelligence without reason”, in J. Mylopoulos and R. Reiter (Eds.) Proceedings of the 12th International Joint Conference on Artificial Intelligence, Morgan Kaufmann, 1991.
- [13] BROOKS, R., “Cambrian Intelligence”, The MIT Press, 1999.
- [14] CAYZER S., AICKELIN U., “A Recommender System based on the Immune Network”, 2002.
- [15] CHIEL, H.J., BEER, R.D., “The brain has a body: Adaptive behavior emerges from interactions of nervous system, body and environment”, Trends in Neurosciences, vol. 20, pp. 553-557, 1997.
- [16] CLIFF, D., HARVEY, I., HUSBANDS, P., “Explorations in evolutionary robotics”, Adaptive Behavior, vol. 2, pp. 73-110, 1993.
- [17] CRESTANI, P.R., VON ZUBEN, F.J., FIGUEIREDO, M.F., “A Hierarchical Neuro-Fuzzy Approach to Autonomous Navigation.” Proceedings of the IEEE International Joint Conference on Neural Networks (*IJCNN'2002*), vol. 3, pp. 2339-2344, in the 2002 IEEE World Congress on Computational Intelligence (*WCCI'2002*), Honolulu, Hawaii, May 12-17, 2002.
- [18] DASGUPTA, D., (Ed.) “Artificial Immune Systems and their Applications”, Springer-Verlag, 1998.
- [19] DAWKINS, R., KREBS, J.R., “Arms races between and within species”, Proceedings of the Royal Society of London B, vol. 205, pp. 489-511, 1979.
- [20] DE CASTRO, L.N., “Engenharia Imunológica: Desenvolvimento e aplicação de ferramentas computacionais inspiradas em sistemas imunológicos artificiais”, dissertação de Doutorado, DCA – FEEC/Unicamp, Campinas/SP, Brasil, Maio de 2001.
- [21] DE CASTRO, L.N., TIMMIS, J.I., “Artificial Immune Systems: A New Computational Intelligence Approach”, Springer-Verlag, 2002.
- [22] DE CASTRO, L. N., VON ZUBEN, F. J., “aiNet: An Artificial Immune Network for Data Analysis”, in Abbass, H. A., Sarker, R.A., Newton, C.S., (Eds.) Data Mining: A Heuristic Approach, Idea Group Publishing, USA, 2001.
- [23] DORIGO, M., COLOMBETTI, M., Robot Shaping: An Experiment in Behavior Engineering (Intelligent Robotics and Autonomous Agents), MIT Press, 1997.

- [24] FARMER, J.D., PACKARD, N.H., PERELSON, A.S., “The Immune System, Adaptation and Machine Learning”, *Physica* 22D, pp. 187-204, 1986.
- [25] FIESLER, E., BEALE, R., “Handbook of Neural Computation”, Institute of Physics Publishing, 1996.
- [26] FIGUEIREDO, M., “Redes Neurais Nebulosas Aplicadas em Problemas de Modelagem e Controle Autônomo”, dissertação de doutorado, DCA – FEEC/Unicamp, Campinas/SP, 1997.
- [27] FIKES, R., NILSON, N., “STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving”, *Artificial Intelligence*, vol. 2, pp. 189-208, 1971.
- [28] FLOREANO, D., MONDADA F., “Evolution of homing navigation in a real mobile robot”, *IEEE Transactions on System, Man and Cybernetics*, 1995.
- [29] FLOREANO, D., MONDADA F., “From evolution of innate behaviors to evolution of learning in robotics agents”, Technical Report R95.06I, Laboratory of Microcomputing, Swiss Federal Institute of Technology at Lausanne, 1995.
- [30] FOGEL, L.J., OWENS, A.J., WALSH, M.J., “Artificial Intelligence Through Simulated Evolution”, Wiley, NY, 1966.
- [31] FORREST, S., HOFMEYR, S.A., SOMAYAJI, A., “Computer Immunology”, *Communications of the ACM*, 40(10), pp. 88-96, 1997.
- [32] FUKUDA T., SHIBATA T., “Fuzzy-Neura-GA-Based Intelligent Robotics”, In J.M. Zurata, R.J.MarksII, C.J. Robinson, (Eds.), *Computational Intelligence Imitating Life*, IEEE Press, 1994.
- [33] GOLDBERG, D.E., *Genetic Algorithms in Search Optimization, and Machine Learning*. Addison-Wesley, Inc., 1989.
- [34] GRENFENSTETTE, J., RAMSEY, J., SCHULTZ, J., “Learning Sequential Decision Rules Using Simulation Models and Competition,” *Machine Learning* 5(4), pp. 355-381, 1990.
- [35] HOLLAND, J.H., “Adaptation in Natural and Artificial Systems”, University Michigan Press, 1975.
- [36] HOLLAND, J.H., *Adaptation in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press, Ann Arbor, MI.91, 1992.

- [37] ISHIGURO, A., ICHIKAWA, S., UCHIKAWA, Y., “A Gait Acquisition of Six-Legged Robot Using Immune Networks,” Proceedings of International Conference on Intelligent Robotics and Systems(*IROS '94*), Munich, Germany, vol.2, pp.1034-1041, 1994.
- [38] ISHIGURO, A., KENDO, T., WATANABE, Y., SHIRAI, Y., UCHIKAWA, H., “Immunoid: A Robot with a Decentralized Consensus-Making Mechanism Based on the Immune System”, Presented at ICMAS Workshop on Immunity-Based Systems, dezembro, pp.82-92 1996.
- [39] JABLONSKI, J., POSEY, J., “Robotics Terminology”, in Handbook of Industrial Robotics, Nof, S., Wiley, J., NY, pp. 1571-1303, 1985.
- [40] JANEWAY, C.A., TRAVERS, P., WALPORT, M., CAPRA, J.D., “Immunobiology: The Immune System in Health and Disease”, 4th edition, Garland Publishing, 1999.
- [41] JERNE, N.K., “The Immune System,” Scientific American, vol.229, N^o. 1, pp. 52-60, 1973.
- [42] JERNE, N.K., “Towards a Network Theory of the Immune System”, Ann. Immunol. (Int. Pasteur) 125C, pp. 373-389, 1974.
- [43] JERNE, N.K., “Idiotypic Networks and other preconceived ideas,” Immunological Rev.,vol.79, pp. 5-24, 1984.
- [44] KOZA, J.R., “Hierarchical Genetic Algorithms Operation on Populations of Computer Programs”, Proceedings of 11th International Joint Conference on Artificial Intelligence, Sao Mateo, CA, Morgan Kaufmann, 1989.
- [45] KOZA, J.R., “Genetic Programming I”, Cambridge, MA, MIT Press, 1992.
- [46] KOZA, J.R., “Genetic Programming II”, Cambridge, MA, MIT Press, 1994.
- [47] KRUISBEEK, A.M., “Tolerance”, The Immunologist, 3/5-6, pp. 176-178, 1995.
- [48] K-TEAM S.A., URL: <http://www.k-team.com>, com informações sobre o robô Khepera, visitado em junho de 2003.
- [49] LANGTON, C.G., (Ed.) “Artificial Life: An Overview”, The MIT Press, 1995.
- [50] LEE, D.-W., JUN H.-B., SIM, K.-B., “Artificial Immune System for Realization of Cooperative Strategies and Group Behavior in Collective Autonomous Mobile Robots,” Proceedings of Fourth International Symposium on Artificial Life and Robotics (*AROB 4th '99*), pp. 232-235, 1999.

- [51] LEE, D.-W., SIM, K.-B., “Artificial Immune Network-based Cooperative Control in Collective Autonomous Mobile Robots,” Proceedings of IEEE International Workshop on Robot and Human Communication, pp. 58-63, 1997.
- [52] MAES, P., “The Dynamics of Action Selection,” Proceedings of Eleventh International Joint Conference on Artificial Intelligence (*IJCAI-89*), Detroit, MI, pp.991-997, 1989.
- [53] MAES, P., “Situated Agents Can Have Goals,” Robotics and Autonomous Systems, vol.6, pp. 49-70, 1990.
- [54] MCFARLAND, D., “Towards Robot Cooperation”, Proceedings of the Simulation of Adaptive Behavior Conference, Brighton, MIT Press, 1994.
- [55] MCFARLAND, D., BOESSER, T., “Intelligent Behavior in Animals and Robots”, MIT Press/Bradford Books, Cambridge, Ma, 1994.
- [56] MICHELAN, R., VON ZUBEN, F.J., Decentralized Control System for Autonomous Navigation based on an Evolved Artificial Immune Network. Proceedings of the 2002 Congress on Evolutionary Computation (*CEC'2002*), vol. 2, pp. 1021-1026, Workshop on Artificial Immune Systems in the 2002 IEEE World Congress on Computational Intelligence (*WCCI'2002*), Honolulu, Hawaii, May 12-17, 2002.
- [57] MILLÁN, J.R., “Rapid, Safe, and Incremental Learning of Navigation Strategies”, IEEE Transactions on System, Man, and Cybernetics-PartB: Cybernetics, vol. 26, n^o. 3, pp.408-420, 1996.
- [58] MITSUMOTO, N., FUKUDA, T., SHIMOJIMA K., OGAWA A., “Micro Autonomous Robotic System and Biologically Inspired Immune Swarm Strategy as a Multi Agent Robotic System” Proceedings of IEEE International Conference on Robotics and Automation, pp.2187-2192, 1995.
- [59] MITSUMOTO, N., FUKUDA, T., ARAI, F., TADASHI, H., IDOGAKI, T., “Self-Organizing Multiple Robotic System,” Proceedings of IEEE International Conference on Robotics and Automation, Minneapolis, MN, pp.1614-1619, 1996.
- [60] MITSUMOTO, N., FUKUDA, T., ARAI, F., ISHIRARA, H., “Control of the Distributed Autonomous Robotic System based on the Biologically Inspired Immunological Architecture” Proceedings of IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, pp.3551-3556, 1997.
- [61] MONDADA, F., FLOREANO, D., “Evolution of neural control structures: Some experiments on mobile robots”, Robotics and Autonomous System, vol. 16, pp. 183-195, 1995.

- [62] NASRAOUI, O., DASGUPTA, D., GONZALEZ, F., “A Novel Artificial Immune System Approach to Robust Data Mining”, In the Proceedings of the Genetic and Evolutionary Computation Conference (*GECCO*), 2002.
- [63] NOLFI, S., “Evolving non-Trivial Behaviors on Real Robots: a garbage collecting robot”, *Robotics and Autonomous System*, vol. 22, pp. 187-198, 1997.
- [64] NOLFI, S., FLOREANO, D., “Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines”, The MIT Press, 2000.
- [65] PARISI, D., “Artificial life and higher level cognition”, *Brain and Cognition*, vol. 34, pp. 160-184, 1997.
- [66] PERELSON, A.S., “Immune Network Theory”, *Immunol. Rev.*, 110:5-36, 1989.
- [67] PERELSON, A.S., OSTER, G.F., “Theoretical Studies of Clonal Selection: Minimal Antibody Repertoire Size and Reliability of Self-NonSelf Discrimination”, *J. theor. Biol.*, 81, pp.645-670, 1979.
- [68] PFEIFER, R., “The Fungus Eater Approach to emotion – A view from Artificial Intelligence”, Technical Report, AI Lab., No. *IFIAI95.04* , Computer Science Department, University of Zurich, 1995.
- [69] POTTER, M., DE JONG, M., GRENFENSTETTE, M., “A Coevolutionary Approach to Learning Sequential Decision Rules,” Proceedings of the Fifth International Conference (*ICGA95*), Pittsburgh, Pennsylvania, 1995.
- [70] RECHENBERG, I., “Evolutionary Strategy”, In J.M. Zurata, R.J.MarksII, C.J. Robinson, (Eds.), *Computational Intelligence Imitating Life*, IEEE Press, 1994.
- [71] RECHENBERG, I., “*Evolutionstrategie: Optimierung Technischer Syetme Nach Prinzipien deer Biologischen Evoltuion*”, Frommann-Holzboog, Stuttgart, 1973.
- [72] ROSENBLATT, J., PAYTON, D., “A Fine-Grained Alternative to the Subsumption Architecture for Mobile Robot Control”, Proceedings of the International Joint Conference on Neural Networks, pp. 317-323, 1989.
- [73] RUSSELL, S., NORVIG, P., “Artificial Intelligence: A Modern Approach”, Second Edition, Prentice Hall, 2003.
- [74] SACERDOTI, E., “Planing in a Hierarchy of Abstraction Spaces”, *Artificial Intelligence*, vol. 5, n° 2, pp. 115-135, 1974.
- [75] SACERDOTI, E., “A Structure for Plans and Behavior”, Ph. D. dissertation, Technical Note N° 109, AI Center, SRI International, Menlo Park, CA, 1975.

- [76] SANTOS J., DURO R.J., BECERRA J.A., CRESPO J.L., BELLAS F., “Considerations in the application of evolution to the generation of robot controllers”, *Journal of Information Sciences*, vol. 133, pp. 127-148, Elsevier Science Inc. 2001.
- [77] SEGEL, N.K., “The Immune System as a Prototype of Autonomous Decentralized Systems”, *Proceedings of Conference on Systems, Man and Cybernetics (SMC’97)*, Orlando, Florida, USA, vol.1, pp. 375-385, 1997.
- [78] STEELS, L., “Exploiting Analogical Representations”, in *Designing Autonomous Agents*, Maes, P. (Ed.), MIT Press, pp.71-88, 1990.
- [79] STEELS, L., “When are robots intelligent autonomous agents?”, *Journal of Robotics and Autonomous Systems*, vol. 15, pp. 3-9, 1995.
- [80] SUSSMAN, G., “A Computer Model of Skill Acquisition”, American Elsevier, N.Y., 1975.
- [81] SUZUKI J., YAMAMOTO Y., “Building an Artificial Immune Network for Decentralized Policy Negotiation in a Communication Endsysten: OpenWebServer/iNexus Study”, 2000;
- [82] TAUBER, I. A., “The Immune Self: Theory or Metaphor?”, New York and Cambridge: Cambridge University Press
- [83] VARELA, F.J., THOMPSON, E., ROSCH, E., “The Embodied Mind: Cognitive Science and Human Experience”, The MIT Press, 1991.
- [84] VARGAS, P.A., DE CASTRO, L.N., MICHELAN, R., VON ZUBEN, F.J., “An Immune Learning Classifier Network for Autonomous Navigation”, in Timmis, J. and Bentley, P. (eds.) *Lecture Notes in Computer Science*, vol. 2787, pp. 69-80, 2003.
- [85] WALTER, W.G., “The Living Brain”, Norton, NY, 1953, (reimpresso 1963).
- [86] WATANABE, Y., ISHIGURO, A., SHIRAI, Y. & UCHIKAWA, Y., “Emergent Construction of Behavior Arbitration Mechanism Based on the Immune System”, In *Proceedings of the Congress on Evolutionary Computation (CEC’98)*, pp. 481-486, 1998.
- [87] WATANABE, Y., ISHIGURO, A. & UCHIKAWA, Y., “Decentralized Behaviour Arbitration Mechanism for Autonomous Mobile Robot Using Immune Network”, In D. Dasgupta (Ed.), *Artificial Immune Systems and their Applications*, Springer, 1999.

- [88] WIENER, N., “Cybernetics, or Control and Communication in Animals and Machines”, Wiley, NY, 1948.