
Arquitetura robótica inspirada na Análise do
Comportamento

Cláudio Adriano Policastro

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito: 15/09/2008

Assinatura: _____

Arquitetura robótica inspirada na Análise do Comportamento

Cláudio Adriano Policastro

Orientadora: *Profa. Dra. Roseli Aparecida Francelin Romero*

Tese apresentada ao Instituto de Ciências Matemáticas e de
Computação - ICMC-USP, como parte dos requisitos para
obtenção do título de Doutor em Ciências - Ciências de
Computação e Matemática Computacional.

**USP – São Carlos
Setembro de 2008**

Dedico este trabalho aos meus pais, Hugo e Amélia, e à minha esposa, Giovana, que tornaram possível este sonho.

Dedico também à meus irmãos, César e Márcio, e à minha nova família Sônia e Luiz, pelo constante apoio e incentivo.

Esta Tese foi preparada com o formatador de textos \LaTeX . O sistema de citações de referências bibliográficas utiliza o padrão *Chicago* do sistema $\text{BIB}\LaTeX$.

© Copyright 2007-2008 por Claudio Adriano Policastro
Todos os Direitos Reservados

Agradecimentos

À minha orientadora, Profa. Dra. Roseli Aparecida Francelin Romero, pela confiança, oportunidade, incentivo, apoio e paciência na orientação deste trabalho e principalmente pela amizade.

Ao Prof. Dr. André Carlos Ponce de Leon Ferreira de Carvalho, pela amizade, incentivo, apoio e oportunidade de iniciar a caminhada que resultou neste trabalho.

Ao Prof. Ednaldo Brigante Pizzolato, pela ajuda e confiança e participação nesta jornada.

Aos amigos do grupo de robótica sociável, Gabriel Dias Pais, Vítor Rocha Munhoz e Renato Ramos da Silva, pelo empenho, dedicação, ajuda e momentos de discussão durante desenvolvimento deste trabalho.

Aos amigo Marcos Quiles e a todos os amigos do Labic e de outros laboratórios, pelo companheirismo, amizade, apoio e momentos de discussão.

Aos professores e funcionários do ICMC, pela eficiência, cordialidade e presteza em diversas ocasiões.

À toda minha família e em especial aos meus pais e minha esposa, pelo incentivo, apoio e dedicação.

À FAPESP, pelo apoio financeiro para a realização deste trabalho.

Resumo

Robôs sociáveis devem ser capazes de interagir, se comunicar, compreender e se relacionar com os seres humanos de uma forma natural. Existem diversas motivações práticas e científicas para o desenvolvimento de robôs sociáveis como plataforma de pesquisas, educação e entretenimento. Entretanto, embora diversos robôs sociáveis já tenham sido desenvolvidos com sucesso, ainda existe muito trabalho para aprimorar a sua eficácia. A utilização de uma arquitetura robótica pode reduzir fortemente o esforço requerido para o desenvolvimento de um robô sociável. Tal arquitetura robótica deve possuir estruturas e mecanismos para permitir a interação social, o controle do comportamento e a aprendizagem a partir do ambiente. Tal arquitetura deve ainda possuir estruturas e mecanismos para permitir a *percepção* e a *atenção*, a fim de permitir que um robô sociável *perceba* a riqueza do comportamento humano e do meio ambiente, e para aprender a partir de interações sociais. Os processos de aprendizado evidenciados na Análise do Comportamento podem levar ao desenvolvimento de métodos e estruturas promissoras para a construção de robôs sociáveis capazes de aprender por meio da interação com o meio ambiente e de exibir comportamento social apropriado. O propósito deste trabalho é o desenvolvimento de uma arquitetura robótica inspirada na Análise do Comportamento. A arquitetura desenvolvida é capaz de simular o aprendizado do *comportamento operante* e os métodos e estruturas propostos permitem o controle e a exibição de comportamentos sociais apropriados e o aprendizado a partir da interação com o meio ambiente. A arquitetura proposta foi avaliada no contexto de um problema real não trivial: o aprendizado da atenção compartilhada. Os resultados obtidos mostram que a arquitetura é capaz de exibir comportamentos apropriados durante uma interação social real e controlada. Ainda, os resultados mostram também que a arquitetura pode aprender a partir de uma interação social. Este trabalho é a base para o desenvolvimento de uma ferramenta para a construção dos robôs sociáveis. Os resultados obtidos abrem muitas oportunidades de trabalhos futuros.

Abstract

Sociable robots should be able to interact, to communicate, to understand and to relate with human beings in a natural way. There are several scientific and practical motivations for developing sociable robots as platform of researches, education and entertainment. However, although several sociable robots have already been developed with success, much work remains to increase their effectiveness. The use of a robotic architecture may strongly reduce the time and effort required to construct a sociable robot. Such robotic architecture must have structures and mechanisms to allow social interaction, behavior control and learning from environment. Such architecture must also have structures and mechanisms to allow *perception* and *attention*, to enable a sociable robot to *perceive* the richness of the human behavior and of the environment, and to learn from social interactions. Learning processes evidenced on Behavior Analysis can lead to the development of promising methods and structures for the construction of sociable robots that are able to learn through interaction from the environment and to exhibit appropriate social behavior. The purpose of this work is the development of a robotic architecture inspired from Behavior Analysis. The developed architecture is able to simulate *operant behavior* learning and the proposed methods and structures allow the control and exhibition of appropriate social behavior and learning from interaction in the environment. The proposed architecture was evaluated in the context of a non trivial real problem: the learning of the shared attention. The obtained results show that the architecture is able to exhibit appropriate behaviors during a real and controlled social interaction. Additionally, the results show also that the architecture can learn from a social interaction. This work is the basis for developing a tool for the construction of social robots. The obtained results open oportunities of many future works.

Sumário

Sumário	iii
Lista de Figuras	vi
Lista de Tabelas	vii
Lista de Algoritmos	ix
Lista de Abreviaturas e Siglas	xi
1 Introdução	1
Introdução	1
2 Robôs Sociáveis	9
2.1 Considerações Iniciais	9
2.2 Projeto de Robôs Sociáveis	10
2.3 Considerações sobre Robôs Sociáveis	14
2.3.1 Atenção e Percepção	14
2.3.2 Auto-Motivação e Emoção	16
2.3.3 Aprendizagem	20
2.3.4 Comportamento	22
2.3.5 Expressão Facial	23
2.3.6 Interação Social	26
2.3.7 Personificação Corporal	28
2.4 Considerações Finais	30
3 Análise do Comportamento	33
3.1 Considerações Iniciais	33
3.2 Comportamento Operante	35
3.3 Modelagem do Comportamento Operante	39
3.4 Discriminação Operante	43
3.5 Privação e Saciação	45
3.6 Considerações Finais	46

4	Arquiteturas Robóticas	49
4.1	Considerações Iniciais	49
4.2	Arquiteturas Deliberativas	51
4.3	Arquiteturas Reativas	55
4.4	Arquiteturas Híbridas	58
4.5	Arquiteturas de Controle de Robôs Sociáveis	62
4.6	Considerações Finais	69
5	Arquitetura Proposta	71
5.1	Considerações Iniciais	71
5.2	Arquitetura Proposta	72
5.3	Técnicas e Algoritmos	75
5.3.1	Aprendizado por Reforço	75
5.3.2	Aprendizado por Reforço Relacional	79
5.3.3	Rede Neural ART 2	82
5.3.4	K-Vizinhos mais Próximos	83
5.4	Sistemas de Apoio	84
5.4.1	Visão Computacional	85
5.4.2	Reconhecimento de Fala	88
5.5	Considerações Finais	90
6	Resultados	93
6.1	Considerações Iniciais	93
6.2	Metodologia de Projeto	94
6.3	Domínios de Aplicação	95
6.3.1	Caixa de Skinner	95
6.3.2	Atenção Compartilhada	97
6.4	Análise dos Resultados	98
6.5	Cabeça Robótica Interativa	100
6.6	Arquitetura Proposta	102
6.6.1	Sistema de Visão Computacional	102
6.6.2	Sistema de Voz	105
6.6.3	Sistema Motor	105
6.6.4	Mecanismo de Aprendizagem por Tutelagem	106
6.6.5	Arquitetura básica	109
6.6.6	Convergência para uma Política Ótima	120
6.7	Simulação da Caixa de Skinner	122
6.8	Simulação da Atenção Compartilhada	130
6.8.1	Simulador de Interações Sociais	130
6.8.2	Principais Resultados	133
6.9	Cabeça Robótica Interativa	144

6.9.1 Experimentos de Aprendizado da Atenção Compartilhada .	144
6.9.2 Experimentos de Aprendizado por Tutelagem	148
6.10 Considerações Finais	151
7 Considerações Finais	153
Referências Bibliográficas	170

Lista de Figuras

2.1	Ciclo de projeto de robôs sociáveis.	13
2.2	Sistema motivacional do robô Kismet.	18
2.3	Arquitetura de um modelo emocional.	19
2.4	Arquitetura de um sistema comportamental.	24
2.5	Expressões faciais do robô Saya.	25
2.6	Expressões faciais do robô Kismet.	25
2.7	A face da robô Vikia.	26
2.8	Expressão facial do robô Valerie.	27
2.9	Estrutura de personificação do robô Kismet.	30
3.1	Distribuições hipotéticas de respostas.	41
4.1	Estratégias de arquiteturas robóticas.	50
4.2	Estrutura da arquitetura SOAR.	52
4.3	Arquitetura PRODIGY.	53
4.4	Arquitetura NASREM.	54
4.5	Arquitetura BLACKBOARD.	56
4.6	Arquitetura SUBSUMPTION.	57
4.7	Arquitetura MOTOR SCHEMA.	58
4.8	Arquitetura AuRA.	60
4.9	Arquitetura DAMN.	61
4.10	Arquitetura baseada em Módulos Situados.	61
4.11	Arquitetura EGO.	63
4.12	Arquitetura do robô Kismet.	65
4.13	Arquitetura SRA.	66
4.14	Arquitetura AD.	67
4.15	Arquitetura do Robô Leonardo.	69
5.1	Arquitetura geral do sistema de controle do robô.	74
5.2	Interação durante aprendizado por reforço.	77

5.3 Episódio de aprendizado do ARR.	81
5.4 Arquitetura da rede ART2	83
6.1 Ciclo de projeto empregado neste trabalho.	95
6.2 Caixa de Skinner com um rato.	96
6.3 Cabeça robótica interativa juntamente com o sistema de controle.	101
6.4 Arquitetura proposta.	103
6.5 Sistema de visão computacional.	104
6.6 Sistema de reconhecimento de fala.	105
6.7 Arquitetura geral do mecanismo de aprendizagem.	107
6.8 Organização geral da arquitetura proposta.	110
6.9 Organização geral do mecanismo de regras de comportamento	112
6.10 Fotografia da memória de trabalho.	113
6.11 Arquitetura do sistema motivacional	115
6.12 Exemplo do processo de aprendizagem de contingências.	119
6.13 Exemplo de encadeamento de comportamentos.	120
6.14 Representação esquemática do simulador da caixa de Skinner.	122
6.15 Influência do sistema motivacional no comportamento do rato	125
6.16 Evolução da aprendizagem durante os experimentos.	129
6.17 Interface do simulador de interações sociais.	131
6.18 Campo visual do robô.	132
6.19 Controle de posicionamento.	132
6.20 Evolução do aprendizado durante os experimentos.	137
6.21 Evolução do aprendizado durante os experimentos.	142
6.22 Evolução do aprendizado durante os experimentos.	147
6.23 Processamento de imagem executado pelo sistema de visão.	148

Lista de Tabelas

6.1	Melhores resultados obtidos durante o processo de aprendizagem.	129
6.2	Resultados do Teste t as melhores médias do PBI.	129
6.3	Melhores resultados obtidos durante o processo de aprendizagem.	136
6.4	Resultados do Teste t as melhores médias do CGI.	137
6.5	Melhores resultados obtidos durante o processo de aprendizagem.	141
6.6	Resultados do Teste t as melhores médias do CGI.	142
6.7	Resultados obtidos em 2.000 unidades de tempo de simulação. .	143
6.8	Resultados do Teste t as melhores médias do CGI.	143
6.9	Resultados obtidos após as 20 corridas de sessão de aprendizado por tutelação.	150

Lista de Algoritmos

1	Aprendizado-Q.	79
2	Versão básica do algoritmo <i>K-Vizinhos mais Próximos</i> para problemas com função objetivo de valores discretos	85
3	Aprendizagem por Contingência.	116

Lista de Abreviaturas e Siglas

AVAM Adaptive View-Base Appearance Models

PCA Análise de Componentes Principais

AR Aprendizado por Reforço

DOF Graus de Libertade

S^D Estímulo Discriminativo Anterior

S^R Estímulo Reforçador

CGI Índice de Olhar Correto

PBI índice de Pressão à Barra

IA Inteligência Artificial

LCA Linguagem de comunicação entre agentes

R Resposta do agente

SCE Sistema de Controle Emocional

SSE Sistema Supervisor Emocional

Introdução

Esta Tese de Doutorado tem como tema a exploração e o desenvolvimento de robôs capazes de interagir com seres humanos de uma maneira natural, denominados robôs sociáveis. Neste capítulo é apresentada uma visão geral do trabalho de pesquisa realizado, bem como os objetivos e um resumo dos resultados obtidos durante a pesquisa. Este Capítulo está organizado da seguinte maneira. Inicialmente são apresentadas as principais motivações para a realização deste trabalho. Então, são mostrados os principais objetivos, o tratamento executado para a consolidação dos mesmos e um resumo dos resultados obtidos. Posteriormente é apresentada a organização desta Tese.

Motivação

Robôs sociáveis são agentes que fazem parte de um grupo heterogêneo, uma sociedade de robôs ou humanos. Estes robôs não devem ser confundidos com sistemas multi-robôs, nos quais diversos robôs devem interagir para a execução de tarefas específicas.

Eles devem poder reconhecer outros robôs e seres humanos e se engajar em interações sociais. Robôs sociáveis devem ser capazes de interagir, de entender, de se comunicar e de se relacionar com seres humanos de uma maneira natural (Breazeal, 2002). Adicionalmente, esses robôs devem ser capazes de aprender a partir das interações com os seres humanos, adquirindo novos conhecimentos e adaptando seus comportamentos em resposta aos estímulos do ambiente (Dautenhahn, 1995) (Dautenhahn and Billard, 1999).

Existem motivações científicas e práticas para o desenvolvimento de robôs sociáveis (Breazeal, 2002). Alguns pesquisadores da área da sociologia (Col-

lins, 1990) (Restivo, 2001) defendem que o conhecimento e a habilidade de se comportar apropriadamente são construções sociais e, portanto, para se desenvolver uma máquina capaz de pensar, adquirir conhecimento e exibir comportamentos apropriados, ela deve fazer parte de nossa sociedade. Então, para um sistema artificial obter sucesso ele deverá ser social.

Pode-se aprender muito sobre a natureza humana com o processo de desenvolvimento de robôs sociáveis (Breazeal, 2002) (Webb, 2000). Robôs sociáveis também são importantes em domínios de problema nos quais os robôs devem interagir com os seres humanos para resolver tarefas específicas, ou em domínios nos quais estes robôs possam ser utilizados como *máquina persuasiva* (Robins et al., 2004) (Scassellati, 2001b) (Björne and Balkenius, 2005). Ainda, nos últimos anos, tem aumentado também o interesse na exploração de robôs sociáveis como robôs de entretenimento com comportamentos semelhantes aos animais (Michaud and Caron, 2000) (Scheeff, 2000) (Kaplan, 2001).

Embora muitos robôs socialmente interativos tenham sido construídos e utilizados com sucesso, ainda existem muitas limitações a serem superadas com o auxílio do desenvolvimento de novas técnicas e a melhoria das já existentes. A maioria dos robôs existentes possui habilidades limitadas de percepção, de cognição e de comportamento, em comparação com seres humanos. O desafio é o desenvolvimento de robôs que possuam a noção de socialização, que possam desenvolver habilidades sociais e que possam mostrar empatia e entendimento do mundo real. Esses robôs ainda representam um objetivo distante e o alcance deste requer a contribuição de outras áreas do conhecimento, como a Psicologia, a Ciência Cognitiva e a Sociologia (Dautenhahn, 1997) (Dautenhahn and Billard, 1999) (Scassellati, 2000).

Para interagir com os seres humanos, os robôs precisam *perceber* e *entender* a riqueza do comportamento humano. Portanto, estes robôs devem possuir sistemas de percepções visuais e auditivas que permita a interação em tempo real, integrando as diversas percepções em uma interface multimodal que possibilite o entendimento e a formação de conceitos sobre os objetos e eventos no ambiente. O desafio aqui é o desenvolvimento de interfaces multimodais que permitam a interação do robô em tempo real, direcionando seus recursos computacionais para o processamento dos estímulos mais relevantes do ambiente (Salichs et al., 2006) (Gockley et al., 2007).

Nas interações humano-robô, outro grande desafio é o desenvolvimento de mecanismos eficientes que permitam ao robô compartilhar com uma pessoa a atenção sobre um objeto ou evento do ambiente, caracterizando uma habilidade denominada *atenção compartilhada*. A atenção compartilhada é um dos grandes desafios a ser solucionado pelos pesquisadores da robótica social (Ka-

plan and Hafner, 2004). Esta habilidade é considerada a base essencial para o desenvolvimento das habilidades sociais e cognitivas (Deák et al., 2001) (Smith and Ulvund, 2003) (Nagai et al., 2003a) (Kanda et al., 2004). Ela foi definida na literatura como a capacidade de utilizar gestos e contato ocular para coordenar a atenção de outras pessoas de forma a compartilhar experiências sobre objetos ou eventos interessantes (Bosa, 2002) (Dube et al., 2004) (Kaplan and Hafner, 2004), possibilitando o aprendizado do que é importante no ambiente (Deák and Triesch, 2005).

Adicionalmente, robôs sociáveis precisam agir pró-ativamente durante as interações com o ambiente. O desenvolvimento de sistemas motivacionais artificiais possibilita que o robô não seja apenas um *escravo* do ambiente e da situação, possibilitando que ele responda a certos estímulos de acordo com suas necessidades internas. Neste contexto, o desafio é o desenvolvimento de sistemas que permitam a simulação de emoções e necessidades, que permitam ao robô interagir de maneira pró-ativa, além de formar a base para o desenvolvimento de comportamentos mais complexos que permitam aos seres humanos determinar o estado atual do robô em relação às suas necessidades (Gadano and Hallan, 2002) (Breazeal, 2003a) (Gockley et al., 2006).

Além da comunicação e da interação, um robô que co-existe diariamente com pessoas deve ser capaz de aprender e se adaptar a novas experiências. Um desafio fundamental é projetar robôs que possam ser ensinados da mesma maneira que outra pessoa ou da forma mais semelhante possível (Dautehahn, 1995) (Breazeal, 2002) (Breazeal, 2004) (Policastro et al., 2007a) (Policastro et al., 2008a).

Se um robô é parte do mundo físico e compartilha o ambiente com os seres humanos, é provável que ele se encontre diariamente com estes. Neste contexto, o desenvolvimento de mecanismo que permitam estabelecer uma longa interação com os seres humanos trás um importante desafio para o projeto de robôs sociáveis (Breazeal, 2004).

Como pode-se verificar pelo exposto, existem diversos desafios para o desenvolvimento de abordagens que permitam aos robôs sociáveis aprender e interagir de uma maneira mais natural. Os processos de aprendizagem e modificação do comportamento, evidenciados na Análise do Comportamento, uma área de Psicologia, podem levar ao desenvolvimento de estruturas e métodos promissores para a construção de robôs sociáveis capazes de aprender a partir das interações com o ambiente e capazes de exibir comportamento social apropriado. Adicionalmente, a utilização de uma representação relacional de primeira ordem (Otterlo, 2005) pode prover um modo econômico para representar o conhecimento necessário a grandes espaços de busca e decisão.

Ainda, como o projeto de robôs sociáveis pode variar muito em termos de

estrutura, técnicas empregadas e objetivos sociais, uma vez que estes robôs podem ser desenvolvidos para diversas aplicações, a organização das estruturas e métodos em uma arquitetura robótica composta por componentes reutilizáveis pode reduzir o tempo e esforço requerido para a construção destes robôs. Tal arquitetura precisa possuir estruturas e mecanismos que permitam a interação social apropriada a partir da aprendizagem e das interações com o ambiente, além de mecanismos que permitam ao robô *perceber* e *entender* a riqueza do comportamento humano.

Objetivos

Apoiado pelas motivações expostas, um dos propósitos deste trabalho de pesquisa é a exploração e o desenvolvimento de estruturas e mecanismos que permitam o controle de robôs sociáveis. Adicionalmente, um objetivo deste trabalho foi a organização destas estruturas e métodos em uma arquitetura robótica com componentes reaproveitáveis que possam ser utilizados para a construção de robôs sociáveis para diversas aplicações. Ainda, este trabalho teve como objetivo secundário a investigação de formas alternativas de representação do conhecimento capazes de representar e manipular grandes espaços de busca e decisão. O desenvolvimento desta pesquisa foi concentrado em alguns aspectos, considerados relevantes para uma contribuição na área da robótica sociável, consubstanciados pelas seguintes questões de pesquisa:

1. Os processos de aprendizado evidenciados na Análise do Comportamento podem levar ao desenvolvimento de estruturas e métodos promissores para a construção de robôs sociáveis?
2. A utilização de uma representação híbrida composta por um mecanismo neural e uma representação relacional de primeira ordem é capaz de representar e manipular grandes espaços de busca e decisão e generalizar o conhecimento adquirido de maneira eficiente?
3. A arquitetura robótica resultante da organização das estruturas e métodos desenvolvidos pode ser um bom modelo para o aprendizado de habilidades sociais básicas como a atenção compartilhada?
4. É possível a construção de um robô sociável simples, utilizando a arquitetura robótica proposta, capaz de realizar uma interação social controlada?

Para o tratamento das questões de pesquisa apresentadas acima, foi realizada uma ampla pesquisa bibliográfica sobre robôs sociáveis, apresentada

no Capítulo 2, seguida de uma pesquisa bibliográfica direcionada com foco específico na teoria da Análise do Comportamento, uma importante ciência inserida na área de conhecimento da Psicologia, apresentada no Capítulo 3. Estas pesquisas possibilitaram a definição dos requisitos e a modelagem lógica das estruturas e métodos que deveriam ser implementados para alcançar os objetivos propostos neste trabalho. Adicionalmente, as pesquisas bibliográficas realizadas possibilitaram a definição de algumas técnicas computacionais para a implementação das estruturas e mecanismos necessários para a realização deste trabalho.

Então, após as pesquisas bibliográficas necessárias ao embasamento teórico, foram realizadas algumas atividades de planejamento e definição da arquitetura proposta. Após estas atividades, a primeira versão da arquitetura foi modelada. Esta versão foi implementada e testada em uma simulação de um *rato* em uma *Caixa de Skinner*, um experimento clássico da Psicologia (Catania, 1999) (Serio et al., 2004).

Embora este experimento preliminar seja simples e possua um espaço de busca e decisão reduzido, estes resultados preliminares demonstram que a arquitetura pode controlar o comportamento de um agente sociável e emitir comportamentos apropriados em uma interação com o ambiente. Os resultados obtidos também permitiram verificar que o sistema motivacional foi capaz de influenciar positivamente no processo de emissão das respostas pelo rato simulado. Portanto os resultados preliminares obtidos mostraram que a primeira versão da arquitetura desenvolvida é uma promissora ferramenta de controle para aplicações na robótica sociável.

Após esta validação, foram verificadas diversas oportunidades de melhorias nos mecanismos e estruturas da arquitetura, de forma a torná-la mais plausível e adequada à teoria na qual ela foi inspirada. Após esta primeira avaliação, também foi realizado um estudo sobre a convergência do algoritmo de aprendizado da arquitetura. A partir deste estudo e da validação, iniciou-se um novo ciclo de planejamento, definição e modelagem para o aperfeiçoamento dos mecanismos existentes e implementação de novos mecanismos na arquitetura. Após isso, a arquitetura foi novamente validada empregando-se o simulador da *Caixa de Skinner* para determinar a eficácia das mudanças realizadas e determinação da capacidade de aprendizado da arquitetura. Os resultados obtidos confirmam os resultados anteriores e mostram que a arquitetura apresentou uma boa capacidade de aprendizado e de exibir comportamentos apropriados durante a simulação. Os resultados obtidos também mostram que o algoritmo de aprendizagem da arquitetura pode constituir uma importante contribuição para a área de aprendizado de robôs sociáveis. Entre suas mais importantes características pode-se citar a sua capacidade de

mapear motivações, estímulos discriminativos e conseqüências para produzir uma política de aprendizagem não determinística.

Após esta nova validação, a arquitetura foi refinada e suas estruturas e métodos foram otimizados. Posteriormente, foram realizados diversos experimentos para a validação da arquitetura em um problema de aplicação real e não trivial, o aprendizado da atenção compartilhada (Dube et al., 2004) (Kaplan and Hafner, 2004). Para tanto, foi desenvolvido um simulador de interações sociais (detalhado na Seção 6.8.1 do Capítulo 6), capaz de simular os movimentos necessários ao robô e ao ser humano, além de fornecer estímulos apropriados do ambiente. Após este estudo, novos mecanismos foram incorporados à arquitetura para possibilitar a generalização do conhecimento adquirido durante o processo de aprendizado. Então, foi realizado um novo estudo para a comparação do desempenho desta nova versão da arquitetura com a versão anterior, além da comparação destas com uma versão da arquitetura empregando o algoritmo Q-Learning clássico como algoritmo de aprendizado.

Os resultados obtidos nestes novos experimentos mostram que a arquitetura é capaz de simular os principais processos básicos de aprendizagem evidenciados na Análise do Comportamento. Os resultados demonstram também que a arquitetura é capaz de exibir comportamentos apropriados durante uma interação social, confirmando os resultados obtidos com os experimentos sobre a Caixa de Skinner. Adicionalmente, os resultados mostram que a arquitetura pode aprender a partir de uma interação social. Portanto, nesta etapa da pesquisa, pôde-se verificar que a arquitetura proposta é uma ferramenta potencial para controlar robôs sociáveis durante interações em um ambiente social. Adicionalmente, os resultados obtidos com estes novos experimentos mostram que a arquitetura proposta é capaz de representar um modelo de aprendizado da atenção compartilhada, sendo uma importante contribuição deste trabalho para a área da robótica sociável.

Finalmente, após este estudo, a arquitetura recebeu um módulo de visão computacional e um módulo de reconhecimento de fala, além de um sistema motor e um mecanismo de aprendizagem por tutela. Após estas últimas extensões, foi realizado um estudo empregando a versão final da arquitetura para controlar uma cabeça robótica interativa em um experimento de aprendizado da atenção compartilhada em um ambiente social real. Os resultados obtidos com estes experimentos confirmam todos os resultados anteriores e mostram que a arquitetura é capaz de exibir comportamentos apropriados durante uma interação social real e controlada. Estes resultados também mostram que a arquitetura proposta é capaz de adquirir habilidades sociais básicas a partir de comportamentos inatos existentes no repertório do robô e da interação com o ambiente.

Adicionalmente, para demonstrar o valor do aprendizado da atenção compartilhada como precursora do aprendizado social, foi realizado um estudo de aprendizado por tutela para validar o novo mecanismo de aprendizagem incorporado a arquitetura. Os resultados mostram que o mecanismo de aprendizagem por tutela pode generalizar o conhecimento instruído. Os resultados obtidos mostram também que a arquitetura proposta é capaz controlar um robô sociável em uma interação controlada, ainda que simples em comparação com muitos trabalhos existentes na literatura. Adicionalmente, os resultados evidenciam que a atenção compartilhada é uma habilidade social fundamental para o desenvolvimento social. Estes resultados permitiram a validação final da arquitetura robótica proposta neste trabalho, assim como a validação de toda a estrutura desenvolvida para possibilitar os experimentos empregando os simuladores e o robô real, permitindo alcançar os objetivos propostos neste trabalho.

O desenvolvimento deste trabalho de pesquisa contribuiu para a formação de uma base para futuros trabalhos na área da robótica sociável. Os resultados obtidos constituem as estruturas fundamentais para a implementação de comportamentos mais complexos e outros mecanismos de raciocínio e aprendizado por meio de interação com o ambiente. Vale aqui ressaltar que até o momento, segundo o conhecimento deste pesquisador, não existem trabalhos similares, na ótica da robótica sociável, sendo desenvolvidos no Brasil.

Organização da Tese

Este trabalho está organizado da seguinte maneira. No Capítulo 2, são introduzidos alguns conceitos e considerações sobre o projeto e o desenvolvimento de robôs sociáveis. No Capítulo 3, são introduzidos alguns dos principais conceitos e processos da Análise do Comportamento, a teoria base deste trabalho. No Capítulo 4, são apresentados os conceitos sobre arquiteturas robóticas e as principais arquiteturas existentes na literatura, além das arquiteturas de controle de robôs sociáveis relacionadas ao presente trabalho. No Capítulo 5, é apresentada a proposta deste trabalho de pesquisa, destacando as diferenças entre este e os principais trabalhos relacionados existentes na literatura, assim como as técnicas computacionais empregadas para o desenvolvimento da arquitetura proposta. No Capítulo 6, são apresentados os mecanismos e estruturas da arquitetura proposta, bem como os resultados obtidos durante os diversos estudos realizados para a validação da mesma. Adicionalmente, são apresentadas algumas discussões específicas sobre os resultados obtidos. Finalmente, no Capítulo 7, são feitas as discussões sobre os resultados, bem como as considerações finais deste trabalho.

Robôs Sociáveis

2.1 *Considerações Iniciais*

Robôs sociáveis são agentes que fazem parte de um grupo heterogêneo, uma sociedade de robôs ou humanos. Estes robôs não devem ser confundidos com sistemas multi-robôs, nos quais diversos robôs devem interagir para a execução de tarefas específicas.

Eles devem poder reconhecer outros robôs e seres humanos e se engajar em interações sociais. Robôs sociáveis devem ser capazes de interagir, entender, se comunicar e se relacionar com seres humanos de uma maneira natural (Breazeal, 2002). Adicionalmente, esses robôs devem ser capazes de aprender a partir das interações com os seres humanos, adquirindo novos conhecimentos e adaptando seus comportamentos em resposta aos estímulos do ambiente (Dautenhahn, 1995) (Dautenhahn and Billard, 1999).

Existem motivações científicas e práticas para o desenvolvimento de robôs sociáveis (Breazeal, 2002). Muitos pesquisadores da área da sociologia defendem que o conhecimento e a habilidade de se comportar apropriadamente são construções sociais e, portanto, para se desenvolver uma máquina capaz de pensar, adquirir conhecimento e exibir comportamentos apropriados, ela deve fazer parte da nossa sociedade. Então, para um sistema artificial obter sucesso ele deverá ser social (Collins, 1990) (Restivo, 2001).

Os robôs sociáveis podem ser utilizados como plataforma de pesquisa, na qual modelos computacionais de habilidades sociais podem ser implementados, testados e analisados por meio da inserção do robô em ambientes sociais controlados. Desta forma, os robôs sociáveis podem ser utilizados por cientistas para estudos e experimentos com o propósito de se entender a natureza

social dos seres humanos (Breazeal, 2002). Adicionalmente, os robôs podem ser utilizados como modelos para testar hipóteses de controle e comportamento de criaturas vivas (Webb, 2000).

Robôs sociáveis também são importantes em domínios de problema nos quais os robôs devem interagir com os seres humanos para resolver tarefas específicas, ou em domínios nos quais estes robôs possam ser utilizados como *máquina persuasiva*. Neste contexto, alguns pesquisadores têm empregado robôs sociáveis como ferramentas de ensino ou como mediadores de interações em terapias com crianças autistas (Robins et al., 2004) (Scassellati, 2001b) (Björne and Balkenius, 2005).

Nos últimos anos, tem aumentado também o interesse na exploração de robôs sociáveis como robôs de entretenimento semelhantes a animais (Michaud and Caron, 2000) (Scheeff, 2000). Neste contexto, pode ser desejável que um robô desenvolva suas habilidades de interação com o passar do tempo. Nesta área, alguns pesquisadores exploram modelos e métodos para ensinar comportamentos complexos aos robôs (Kaplan, 2001).

Embora muitas pesquisas sobre robôs sociáveis estejam sendo desenvolvidas com sucesso, ainda existem muitas limitações a serem superadas. A maioria dos robôs existentes possuem habilidades limitadas de percepção, de cognição e de comportamento. Como sistemas especialistas, entretanto, é possível o desenvolvimento de robôs altamente sofisticados que atuam em áreas com socialização restrita (Breazeal, 2000). O desafio é o desenvolvimento de robôs que possuam a noção de socialização, que possam desenvolver habilidades sociais e que possam mostrar empatia e entendimento do mundo real. Esses robôs ainda representam um objetivo distante e o alcance deste requer a contribuição de outras áreas como Psicologia, Ciência Cognitiva e Sociologia (Dautenhahn, 1997) (Dautenhahn and Billard, 1999) (Scassellati, 2000).

Neste Capítulo, são introduzidos os principais conceitos sobre robôs sociáveis, apresentando algumas considerações sobre abordagens, teorias e técnicas utilizadas para o desenvolvimento destes robôs. Este Capítulo está organizado da seguinte maneira. Na Seção 2.2, são apresentados brevemente alguns desafios a serem superados durante a realização de projetos de robôs sociáveis. Na Seção 2.3, são discutidas algumas questões sobre o desenvolvimento de robôs sociáveis. Finalmente, na Seção 2.4, são feitas as considerações finais deste Capítulo.

2.2 Projeto de Robôs Sociáveis

Nesta Seção, são apresentados alguns desafios a serem superados durante a execução de projetos para a construção de robôs sociáveis. O sucesso da

interação entre um robô sociável e os seres humanos não depende somente dos mecanismos e estruturas do robô, mas também da sua robustez em termos de membro da sociedade (Sabanovic et al., 2007). A incorporação de princípios relevantes do comportamento social humano e o entendimento do contexto sócio-cultural no qual se darão as interações do robô em sociedade poderão influenciar fortemente os resultados do seu projeto. Um robô sociável somente será aceito em uma sociedade de humanos se este apresentar comportamentos sociais adequados, incluindo expressões faciais, gestos, atenção e voz (Breazeal, 2003b) (Breazeal, 2004). Durante uma interação, tanto o robô quanto ser humano devem compartilhar o controle da tarefa em questão. Os robôs devem controlar e permitir o controle de uma conversação, alternando entre fala e atenção sobre o parceiro social ou, ainda, segundo Skinner (Skinner, 1953), pode-se supor interações adequadas a partir do ponto de vista do comportamento verbal, no qual alternam-se os papéis de falante e de ouvinte.

O projeto de robôs sociáveis pode ser visto como a força motora para a criação de uma nova *Ciência Híbrida* (Sabanovic et al., 2007). Estes projetos devem envolver não somente engenheiros e cientistas da computação, mas também cientistas sociais, psicólogos, desenhistas, entre outros. Em alguns projetos, dependendo da aplicação do robô, um time de outros profissionais pode ser incluso na equipe de desenvolvimento, como médicos, pedagogos e professores (Robins et al., 2004) (Scassellati, 2001b) (Björne and Balkenius, 2005) (Sabanovic et al., 2007).

A produção de conhecimento durante a execução de um projeto de um robô sociável não deve ser um desafio individual para cada membro da equipe, mas sim um resultado de um esforço conjunto em um trabalho de colaboração entre todos os membros desta equipe. Portanto, o projeto de um robô sociável demanda a definição de uma metodologia e uma linguagem comum de trabalho, para possibilitar a coordenação e a evolução da equipe de projeto, assim como para um entendimento comum sobre o projeto que está sendo desenvolvido (Sabanovic et al., 2007).

Durante a execução de um projeto de um robô, sociável ou não, são necessárias as resoluções de problemas que incluem desde o mecanismo de cognição (planejamento, tomada de decisão, aprendizado social e conceitual, incorporação), percepção (navegação, sensibilidade do ambiente), interação com os seres humanos (interface, dispositivos de entrada) até a arquitetura (sistema de controle e sistema eletromecânico). Adicionalmente, no caso de um robô sociável, é necessário a resolução de problemas adicionais impostos pela interação social (Dautenhahn, 1998) (Breazeal, 2000) (Breazeal, 2002) (Deák et al., 2007).

Uma abordagem recentemente empregada para o projeto de robôs sociáveis

é a *biologicamente inspirada*. Nesta abordagem, os pesquisadores objetivam criar robôs que imitam ou simulam o comportamento social ou inteligente encontrado nas criaturas vivas. Projetos biologicamente inspirados são baseados em teorias da neurociência e ciências biológicas, incluindo antropologia, psicologia, etiologia, sociologia, entre outras. Estas teorias tem sido largamente empregadas para direcionar o projeto dos sistemas motivacional, motor, cognitivo e comportamental do robô (Dautenhahn, 1997) (Dautenhahn, 1998) (Dautenhahn, 2000) (Breazeal, 2000).

Embora o projeto de robôs sociáveis possa variar em termos de estrutura e técnicas empregadas e objetivos sociais, uma vez que estes robôs podem ser desenvolvidos para diversas aplicações, um exame da literatura permite identificar alguns elementos comuns a estes projetos. Muitos pesquisadores ressaltam a importância da validação do robô sociável em um ambiente real como forma de aprimorar o projeto do mesmo (Breazeal, 2004) (Sidner et al., 2005) (Sabanovic et al., 2006). Um exame destes trabalhos permite a suposição de uma metodologia de execução de projeto segundo a qual o robô possa ser desenvolvido de forma incremental e validado a cada ciclo de desenvolvimento, de forma a ter suas funcionalidades melhoradas até que os objetivos do projeto sejam alcançados.

Um possível processo de projeto de robôs sociáveis pode ser definido como uma espiral na qual o robô pode ser desenvolvido de maneira incremental a cada ciclo da mesma. Nas etapas iniciais do projeto, uma prova de conceito e um protótipo do robô podem ser desenvolvidos para que a equipe efetue os experimentos e validações iniciais. Então, um projeto mais detalhado pode ser elaborado e, de maneira incremental, novas estruturas e métodos podem ser incorporados ao robô, de forma que este tenha todas as funcionalidades desejadas para o atendimento dos objetivos do projeto.

Inicialmente o projeto deve passar por uma etapa de *análise*. Durante esta etapa, o domínio da aplicação deve ser descrito e identificado para ser melhor compreendido. Os objetivos do projeto devem ser delineados e devem ser definidas a equipe de desenvolvimento e as ferramentas a serem utilizadas para construir o robô.

Após a *análise* do problema, o robô pode ser desenvolvido de maneira incremental em ciclos de processo compostos por quatro fases, *planejamento*, *projeto*, *construção*, *observação e validação*. Durante o *planejamento*, a equipe de projeto deve realizar o planejamento das etapas do projeto, bem como os itens a serem desenvolvidos e os recursos necessários para cada uma. Durante o *projeto*, a equipe de projeto deve empregar processos de engenharia, entre outros, para a modelagem dos entregáveis da fase sendo executada (documentos, especificações, softwares, peças, componentes, etc). Durante a

construção, a equipe de projeto deve empregar as técnicas selecionadas para a construção dos entregáveis da fase sendo executada. Durante a *observação e validação*, a equipe deve submeter o robô a interações em ambiente real para avaliar os problemas de projeto, o comportamento do robô e as próximas etapas do projeto. Esta análise da interação social do robô, assim como de seu comportamento, pode fornecer elementos relevantes para o aprimoramento do projeto do robô e levar a um novo ciclo do processo. Ainda, em relação à fase observação e validação, Sabanovic (Sabanovic et al., 2007) ressalta a importância da validação do robô em um ambiente social externo ao laboratório, em interações com pessoas externas à equipe do projeto. Entretanto, deve-se esclarecer que esta interação somente pode ser realizada em estágios mais avançados do processo.

Nos ciclos iniciais da espiral (ver Figura 2.1), uma prova de conceito e um protótipo podem ser desenvolvidos para que a equipe efetue os experimentos e validações iniciais. Após isso, a cada ciclo uma nova funcionalidade pode ser incorporada ao robô, de forma que este tenha todas as funcionalidades desejadas de forma incremental.

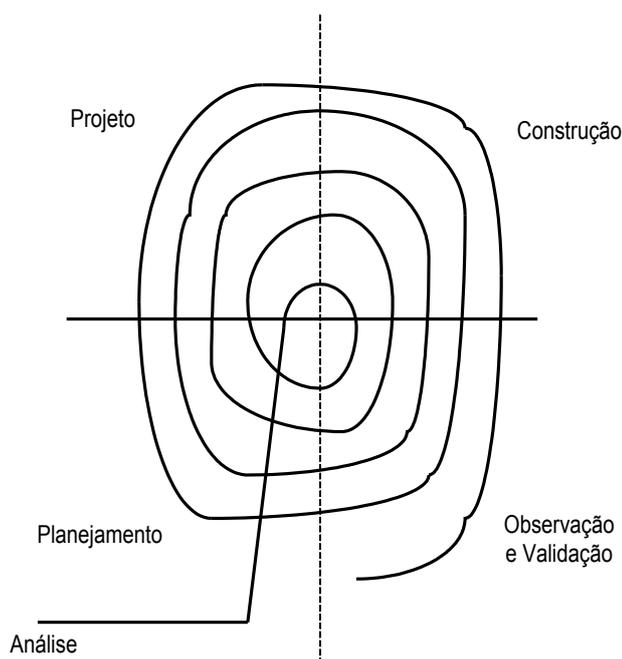


Figura 2.1: Ciclo de projeto de robôs sociáveis. Durante a *análise* a equipe de projeto deve delinear os objetivos do projeto e as técnicas que serão empregadas na construção do robô. Após isso, o processo de projeto inicia uma espiral que passa pelas fases de *planejamento*, *projeto*, *construção*, *observação e validação* a cada ciclo do processo. Nos ciclos iniciais da espiral, uma prova de conceito ou protótipo pode ser desenvolvido para que a equipe efetue os experimentos e validações iniciais. Posteriormente, a cada ciclo uma nova funcionalidade pode ser incorporada ao robô, de forma que este tenha todas as funcionalidades desejadas de forma incremental.

2.3 Considerações sobre Robôs Sociáveis

O desenvolvimento de agentes sociais precisa ser balanceado em termos de cognição, aquisição de experiências, credibilidade e incorporação, entre outros aspectos. Nesta Seção, são apresentadas algumas considerações sobre o desenvolvimento de mecanismos e estruturas para robôs sociáveis (Dautenhahn, 1998) (Breazeal, 2001) (Breazeal, 2002).

2.3.1 Atenção e Percepção

Para interagir com os seres humanos, os robôs precisam *perceber* e *entender* o comportamento humano (Scassellati, 2001a) (Breazeal, 2002).

Especificamente, os robôs sociáveis precisam possuir uma percepção orientada aos seres humanos, ou seja, otimizada para interagir com estes (Sidner et al., 2004). Eles devem ser capazes de rastrear características humanas (faces, mãos, corpo), além de interpretar a fala e a linguagem natural. Adicionalmente, eles devem incorporar mecanismos para o reconhecimento de expressões faciais, gestos e atividades humanas (Breazeal and Scassellati, 1999) (Breazeal, 2000) (Scassellati, 2001a) (Aggarwal and Cai, 1999) (Goodrich and Schultz, 2007).

Recentemente, diversas abordagens de sistemas de percepção para robôs sociáveis foram propostas na literatura. Em (Breazeal and Scassellati, 1999) (Breazeal, 2002), por exemplo, é apresentado um sistema de atenção que integra percepções visuais: detecção de movimento; cores e faces humanas, juntamente com efeitos de habituação. Cada percepção visual gera uma mapa de características que são, então, combinados por meio de uma soma ponderada. Esse sistema influencia e é influenciado por um sistema de comportamento e por um sistema motivacional, provendo um sistema de atenção dependente do contexto do ambiente e da situação e necessidade do robô. Desta forma, o sistema combina as diversas características providas pelo sistema perceptual juntamente com os estados atuais de motivação e comportamento para direcionar os limitados recursos computacionais durante as interações sociais.

Para que um robô sociável seja capaz de interagir com seres humanos de maneira natural, este também deve possuir mecanismos que permitam o reconhecimento e a riqueza das informações transmitidas pela voz das pessoas. O reconhecimento de voz é geralmente efetuado em um processo de duas etapas: o processamento para transformar o sinal de áudio em um vetor de características, seguido pela busca do vocabulário em uma base de conhecimento que contém a gramática a ser reconhecida. Muitos sistemas atuais de reconhecimento de voz utilizam *Modelos de Markov* para determinar o vocábulo mais semelhante ao sinal recebido (Rabiner and Juang, 1993). Em diversas

aplicações, um sistema de reconhecimento deve ser capaz de reconhecer as diversas dimensões da fala: quem está falando, o que está sendo falado e como está sendo falado. Estas características permitem ao robô efetuar uma busca pela pessoa que está falando, ou iniciar um diálogo, ou analisar as emoções contidas na fala (Breazeal, 2000) (Lauria et al., 2002).

Adicionalmente, os robôs sociáveis devem ser capazes de integrar as percepções visuais e auditivas, a partir de uma interface multimodal, para a formação de conceitos e complemento das informações, uma vez que os seres humanos freqüentemente utilizam gestos para complementar a fala e para compactar informações geométricas (localização, direção de movimento) (Lorenz, 1998) (Wu and Huang, 1999) (Waldherr et al., 2000).

Na Literatura foram propostos muitos robôs que integram interfaces multimodais, integrando visão, voz e personificação corporal, como por exemplo o robô *Maggie* (Salichs et al., 2006) e o robô *Grace* (Gockley et al., 2007).

Em (Salichs et al., 2006) é apresentado um robô denominado Maggie que permite a interação multimodal, utilizando toques, expressões faciais e corporais, além de comunicação verbal. A base de Maggie é equipada com duas rodas e dois motores com *encoders*. A base também é equipada com 12 para-choques, 12 sensores ópticos infravermelhos e 12 sensores de ultra-som. Sobre a base, existe um sensor laser para buscas e mapeamento do ambiente. A parte superior do robô incorpora os mecanismos de interação. Sobre a plataforma, foi adicionada uma cabeça com aspectos antropomórficos com uma aparência atraente. A cabeça possui dois graus de liberdade (DOF) permitindo movimentos básicos no sentido vertical e horizontal. A cabeça possui dois olhos negros, uma boca, câmeras ocultas nos olhos e duas pálpebras móveis e controláveis. Maggie possui dois braços com 1 DOF para prover expressões não verbais por movimento do corpo. Ainda, um sistema de voz permite que o robô fale e reconheça fala em Castelhana. Este robô possui também uma câmera para localizar e rastrear pessoas próximas a ele.

Em (Gockley et al., 2007) é apresentado um robô denominado Grace. Este robô possui um corpo com um monitor LCD que exibe uma face expressiva renderizada em 3D. Ele é capaz de fala por meio de um sintetizador de voz e a face do robô movimentada os lábios sincronamente com a fala. O robô também utiliza um sensor laser que mapeia o ambiente e rastreia as pessoas por um método de segmentação.

Nas interações humano-robô, outro grande desafio é o desenvolvimento de mecanismos eficientes que permitam ao robô compartilhar com uma pessoa a atenção sobre um objeto do evento do ambiente, caracterizando uma habilidade social denominada *atenção compartilhada*. A atenção compartilhada é um dos grandes desafios a ser solucionado pelos pesquisadores da robótica

social (Kaplan and Hafner, 2004). Esta habilidade é considerada a base essencial para o desenvolvimento das habilidades sociais e cognitivas (Deák et al., 2001) (Smith and Ulvund, 2003) (Nagai et al., 2003a) (Kanda et al., 2004). Ela foi definida na literatura como a capacidade de utilizar gestos e contato ocular para coordenar a atenção de outras pessoas de forma a compartilhar experiências sobre objetos ou eventos interessantes (Bosa, 2002) (Dube et al., 2004) (Kaplan and Hafner, 2004), possibilitando o aprendizado do que é importante no ambiente (Deák and Triesch, 2005).

Na literatura foram propostos diversos mecanismos da atenção compartilhada que permitem ao robô adquirir a habilidade da atenção compartilhada por meio de interações com o ambiente.

Em um importante trabalho ainda em andamento, por exemplo, (Fasel et al., 2002) (Carlson and Triesch, 2003) (Lau and Triesch, 2004) (Deák and Triesch, 2005) é apresentado um modelo desenvolvimentista da atenção compartilhada baseado na interação entre motivações inatas e aprendizado de contingências. O modelo possui um conjunto básico de preferências afetivo-motivacionais e habilidades de processamento de percepções, um mecanismo de aprendizado por reforço e uma estrutura do ambiente social que provê informações apropriadas para o aprendizado da habilidade da atenção compartilhada. Uma suposição fundamental é que os processos de aprendizado do cérebro infantil se ajustam aos padrões da seqüência de eventos durante as interações sociais. O conjunto básico é composto por: mecanismos de percepções e motivações que simulam as preferências das crianças e direciona o olhar para estímulos interessantes do ambiente, um mecanismo de habituação que progressivamente diminui o estímulo de um objeto do ambiente, estratégias de aprendizado (adaptação) de comportamentos para satisfazer os mecanismos de percepções citados, um ambiente estruturado provendo correlações entre o olhar de um adulto e a localização de um estímulo interessante.

2.3.2 Auto-Motivação e Emoção

Robôs sociáveis precisam agir pró-ativamente durante as interações com o ambiente. Para tanto, estes devem possuir mecanismos que levem a esta pró-atividade. Neste contexto, um sistema motivacional possibilita que o robô não seja apenas um *escravo* do ambiente e da situação, possibilitando que ele responda a certos estímulos de acordo com suas necessidades internas. Esse sistema pode servir como base para a construção de mecanismos de interação mais complexos, ativando comportamentos específicos para determinados estímulos e contextos do ambiente. Muitos destes sistemas motivacionais incorporaram, ao menos, mecanismos emocionais rudimentares (Gockley et al., 2006). As emoções são um importante sistema motivacional e possuem um

papel importante no comportamento, na comunicação e na interação entre seres humanos (Ekman, 1999) (Breazeal and Scassellati, 2000) (Gadano and Hallan, 2002) (Breazeal, 2003a).

Diversos pesquisadores argumentam que existe um pequeno conjunto de emoções, denominadas básicas ou primárias, que foram preservadas durante o processo de evolução, pois elas possibilitam a adaptação do comportamento dos organismos a diversas situações com as quais as criaturas se deparam diariamente. Entre as apresentadas na literatura como básicas, pode-se citar a raiva, a aversão, o medo, a alegria, a tristeza e a surpresa. Estas emoções servem a propósitos específicos e emergem em determinados contextos do ambiente, para preparar o organismo a responder apropriadamente (Ekman, 1999) (Breazeal, 2003a).

As emoções podem ser utilizadas para determinar o controle entre modos de comportamentos, coordenando ações, aprendizado e adaptação (Gadano and Hallan, 2002). Muitos pesquisadores investigaram o uso de emoções em interações entre robôs e seres humanos (Gadano and Hallan, 2002) (Breazeal, 2003a) (Gockley et al., 2006).

Em (Breazeal, 2000) (Breazeal and Scassellati, 2000) (Breazeal, 2002) (Breazeal, 2003a) é apresentado um sistema motivacional que é composto por dois subsistemas: um sistema que implementa um conjunto de necessidades e um sistema que implementa emoções e estados de expressões. O sistema de necessidades influencia a seleção do comportamento do robô e o comportamento expressivo do robô. O sistema de emoções e estados de expressão influencia as expressões do robô e regula as interações com seres humanos. O sistema motivacional e emocional é capaz de prover um retorno do estado interno do mesmo, possibilitando que os seres humanos interpretem as necessidades do robô e as satisfaçam. No modelo apresentado, as motivações do robô são compostas por necessidades e emoções, estabelecendo suas necessidades e influenciando como e quando o robô deve agir para satisfazê-las. As emoções e necessidades de Kismet são projetadas de tal forma que o robô mostre um comportamento positivo quando estiver em uma interação saudável com pessoas e demonstre emoções negativas quanto a interação estiver super ou sub estimulando o robô. Em conjunto com as necessidades do robô, o sistema emocional é projetado para ser um sistema flexível que coordena e recebe informações sobre estímulos do ambiente e estados internos para produzir uma resposta que atenda ao propósito de comportamento social apropriado, além de manter o estado interno do robô em níveis aceitáveis. As emoções podem ser ativadas por diversos eventos que são avaliados como sendo significantes ao “bem estar” do robô. Uma vez ativa, uma emoção serve a um conjunto particular de funções ou comportamentos que poderão atuar para estabelecer

a relação desejada entre o robô e seu ambiente. As emoções motivam o robô a entrar em contato com coisas que promovem seu “bem estar” e a evitar o contato com coisas que não lhe fazem bem. Essa implementação é fortemente inspirada em modelos da etologia sobre percepção, motivação, e comportamento. Na Figura 2.2 é ilustrada a arquitetura deste sistema, exemplificando o processamento do medo.

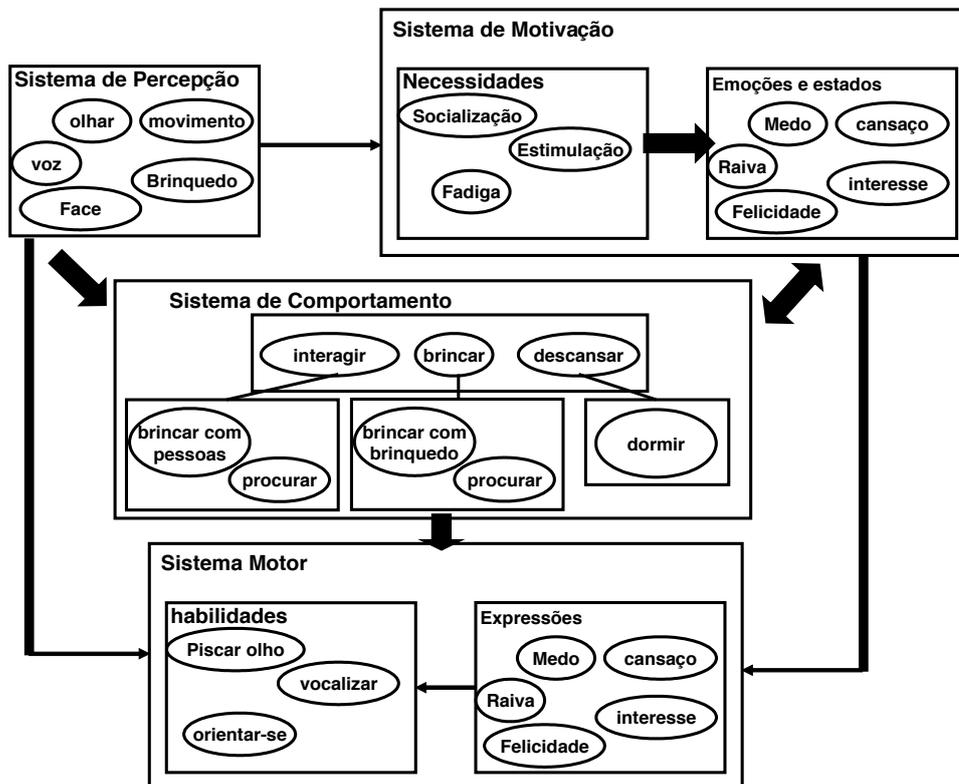


Figura 2.2: Sistema motivacional do robô Kismet. As setas indicam o fluxo de informações entre os módulos. Quando o robô detecta algum estímulo no ambiente, o sistema de percepção o codifica e o envia para o sistema de comportamento e para o sistema motivacional. A informação sobre os estímulos é filtrada pelo extrator correspondente de cada processo de emoção e necessidade. Como resultado, o sistema motivacional passa sua ativação a ao sistema de comportamento que ativa o comportamento emotivo correspondente. Isto gera uma alteração na expressão facial, postura de corpo, e qualidade vocal (Breazeal, 2003a).

Em (Gadano and Hallan, 2002), um modelo emocional é proposto, na forma de uma rede neural recorrente, no qual as emoções dependem e influenciam a percepção do ambiente. Esse modelo é integrado a uma arquitetura de aprendizado por reforço com três diferentes propósitos: influenciar as percepções, prover reforço e determinar quando uma situação deve ser reavaliada. O modelo foi implementado baseado em 4 emoções: felicidade, tristeza, medo e raiva. A intensidade de cada emoção é determinada pelos sentimentos internos correntes, definido como sensações e percepções. Entretanto, o es-

tado emocional influencia os sentimentos internos do modelo por meio de um sistema hormonal, produzindo um hormônio apropriado. O sistema hormonal é simples e consiste de um hormônio associado a cada sentimento. A quantidade de hormônio produzido por cada emoção é diretamente relacionada a sua intensidade e ela depende dos sentimentos internos. Esse modelo, integrado ao sistema de aprendizado, é utilizado para direcionar o comportamento do robô e o aprendizado do mesmo de acordo com os estímulos do ambiente. Na Figura 2.3 é ilustrada a arquitetura deste sistema emocional.

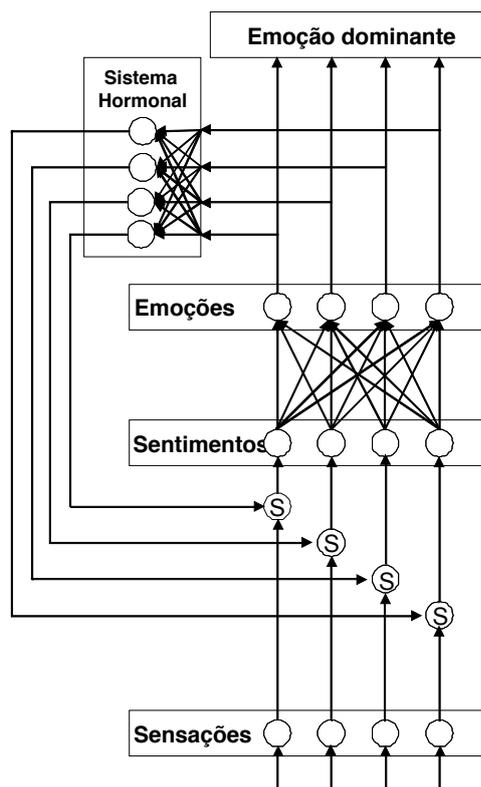


Figura 2.3: Arquitetura de um modelo emocional. Neste modelo, as emoções não dependem diretamente da percepção imediata do agente do mundo, ou seja, suas sensações. Eles dependem dos sentimentos que são uma combinação das sensações e dos hormônios produzidos recentemente pelo sistema hormonal em resposta às emoções ativas. Isto acrescenta ao estado de emoção alguma memória sobre o passado recente (Gadano and Hallan, 2002).

Em (Gockley et al., 2006), um modelo afetivo para robôs sociáveis é apresentado. Este modelo possui um subconjunto básico de emoções sugeridas por Ekman (Ekman, 1999). Estas emoções são: felicidade, tristeza, frustração, e raiva. Cada emoção é associada a um nível de intensidade, representado como um número real que varia entre 0 (não existente) e 1 (intensidade mais alta), como também uma taxa de valência (positivo ou negativo). Para cada emoção, são definidas expressões de diferentes intensidades que podem ser exibidas pela face do robô, de acordo com a intensidade da ativação da emo-

ção. Este modelo é integrado ao sistema de reconhecimento de voz, de forma que certas declarações podem disparar emoções específicas diretamente, como elogios que causam felicidade ou insultos que resultam em tristeza. Neste modelo, as emoções não acontecem simultaneamente e nem são utilizadas para compor uma nova emoção complexa.

2.3.3 *Aprendizagem*

Além da comunicação e da interação, um robô que co-existe diariamente com pessoas deve ser capaz de aprender e se adaptar a novas experiências. Idealmente, as pessoas deverão poder ensinar para o robô como executar novas tarefas ou detalhes de como executar uma determinada tarefa. Conseqüentemente, um desafio fundamental é projetar robôs que possam ser ensinados da mesma maneira que outra pessoa ou da forma mais semelhante possível (Dautenhahn, 1995) (Breazeal, 2002) (Breazeal, 2004).

Entretanto, existem algumas questões a serem consideradas durante o projeto dos mecanismos de aprendizagem. Frente a grande complexidade e número de estímulos que chegam aos sistema de percepção de um robô, este precisa decidir quais estímulos são relevantes para o seu aprendizado. A determinação de quais estímulos são relevantes ao processo de aprendizado pode ser entendida como uma questão de saliência (Itti et al., 1998) (Breazeal, 2002) (Breazeal, 2004). A saliência dos estímulos pode ser determinada internamente, por meio do processamento das propriedades dos estímulos (cor, tamanho, orientação, proximidade, entre outras), ou externamente, apontadas por um professor.

Uma vez que os estímulos salientes foram determinados e identificados, o robô deve ser capaz de determinar qual a ação a ser tomada no presente contexto do ambiente (Breazeal, 2002) (Breazeal, 2004). Conforme aumentam as funcionalidades do robô, aumenta seu repertório de possíveis ações. Este fato também contribui para um aumento no espaço de busca por ações apropriadas. A determinação de qual ação deve ser tomada pode ser efetuada de diversas maneiras. O robô pode selecionar ações e experimentá-las em um processo de aprendizado por contingência, ou o robô pode selecionar suas ações baseado em suas experiências prévias. Na literatura foram propostas diversas abordagens de aprendizagem para robôs sociáveis (Marom and Hayes, 2001) (Nagai et al., 2003a) (Lockerd and Breazeal, 2004) (Gold and Scassellati, 2007).

Lockerd e Breazeal (Lockerd and Breazeal, 2004), por exemplo, apresentam um mecanismo de aprendizagem, implementado em um robô humanóide, para demonstrar que um diálogo colaborativo pode permitir a um robô aprender uma tarefa por meio da tutelação de um ser humano. O robô possui sistemas

de visão e reconhecimento de fala para permitir a interação multimodal com o mesmo. O sistema cognitivo recebe dados continuamente destes sistemas de visão e fala, e os integra para formar diversas convicções sobre objetos do mundo, assim como sobre gestos e fala dos seres humanos.

Em (Marom and Hayes, 2001), um mecanismo de imitação biologicamente inspirado é apresentado, baseado em uma estrutura denominada *mirror neurons* ou neurônios espelho. Estudos em primatas sugerem que essa estrutura é responsável pela imitação de comportamentos e gestos observados em outros agentes. O mecanismo é composto por um sistema perceptual e por uma rede de esquemas que simula as funções dos neurônios espelho. O sistema de percepção é formado por uma rede neural auto-organizável de mapa de características que é capaz de reconhecer e classificar os estímulos do ambiente. A rede de esquemas é formada por esquemas de percepção e esquemas de motor. Os esquemas são módulos independentes que quando ativados processam informações perceptuais ou motoras. Cada esquema de percepção mantém uma estrutura que pode reconhecer uma seqüência temporal de estímulos do ambiente. Cada esquema de motor mantém uma estrutura que gera uma seqüência motora que resulta em uma parte de um comportamento. Quando um esquema de percepção da rede recebe um estímulo, ele o compara com a estrutura armazenada para produzir uma medida de confiança. Se essa medida é suficientemente alta (acima de um *limiar*), então o correspondente esquema motor é ativado.

Em (Gold and Scassellati, 2007), é apresentado um robô que aprende a mapear a relação entre sua própria atividade motora e os movimentos realizados, por meio da construção de um modelo *Bayesiano*. Após aprender este modelo, que mapeia seus movimentos a seus comandos motores, o robô emprega o mesmo para construir um modelo para entender os movimentos de outros organismos. Este modelo do movimento de outros agentes pode ser visto como uma cópia de seu próprio modelo de movimento, mas com as atividades motoras não explícitas. O robô emprega três modelos construídos para todos os objetos em seu campo visual e utiliza uma Rede Neural Bayesiana (Haykin, 1999) para determinar quando utilizar estes modelos a partir dos estímulos percebidos no ambiente. O modelo de “objetos inanimados” é o mais simples, no qual assume-se que objetos inanimados podem ter movimento somente devido a ruídos dos sensores ou quando eles são manipulados. O segundo modelo do robô é o “auto-modelo” no qual as ações motoras do robô são mapeadas aos movimentos observados do mesmo. O terceiro modelo é o que explica o movimento de outras pessoas (ou outros agentes animados) no campo visual. Este modelo é idêntico ao auto-modelo. Entretanto, neste modelo as ações motoras são inferidas pela observação dos movimentos dos

outros agentes, a partir do mapeamento definido pelo segundo modelo (auto-modelo).

Em Nagai (Nagai et al., 2003a) (Nagai et al., 2003b) é apresentado um modelo construtivo pelo qual o robô adquire a habilidade da atenção compartilhada baseado em um mecanismo da atenção visual e aprendizado com auto-avaliação. O modelo apresentado é composto por: um módulo de atenção visual, um controle de retorno visual, um módulo de aprendizagem com um avaliador interno e um controle que produz uma saída composta pela seleção da saída do sistema de atenção visual ou do módulo de aprendizagem. O sistema de atenção visual é capaz de detectar: cores, bordas, faces e movimentos. O controle de retorno visual recebe as características detectadas do objeto i_n e gera um comando de motor para direcionar a cabeça do robô para o objeto. Esse comando de motor é gerado pelo mapeamento entre a posição (x, y) do objeto na imagem I e os ângulos vertical e horizontal da cabeça do robô, que é então enviado para o controle de saída. O avaliador interno verifica se a distância entre o centro da imagem capturada pela câmera e o centro de um objeto detectado é menor que um limiar definido previamente, para determinar o sucesso do sistema visual em localizar um objeto. O módulo de aprendizado é formado por uma rede neural de 3 camadas que recebe como entrada a imagem capturada pela câmera do robô e os ângulos vertical e horizontal da câmera. No modo de aprendizado, esta rede é treinada utilizando o algoritmo *back propagation*. Essa rede neural produz como saída os comandos de motor para direcionar a cabeça do robô para o objeto. Esse comando de motor também é enviado para o controle de saída, que arbitra entre a utilização dos comandos de motor do sistema visual ou do sistema de aprendizado. Inicialmente, enquanto o sistema está aprendendo, o controle de saída atribui uma alta prioridade para a seleção dos comandos de motor produzidos pelo sistema visual. Nos estágios posteriores, o controle de saída atribui, gradualmente, uma prioridade maior para a seleção dos comandos de motor produzidos pelo sistema de aprendizado.

2.3.4 Comportamento

Seres humanos são atraídos por agentes que demonstram comportamentos similares aos das criaturas vivas. Eles possuem uma tendência a animar e a antropomorfizar os agentes da natureza e os artefatos tecnológicos. Antropomorfismo é a tendência de atribuir características humanas a objetos com o objetivo de auxiliar a racionalizar suas ações (Duffy, 2003). Seres humanos tendem a interpretar comportamentos como sendo intencionais, mesmo quando interagem com agentes artificiais (Duffy, 2003). Para obter êxito nas interações com os seres humanos, os robôs sociáveis precisam exibir compor-

tamentos que possibilitam às pessoas atribuir intenções e desejos, permitindo assim uma interação natural entre os robôs e os seres humanos (Breazeal, 2002). Os robôs sociáveis precisam exibir comportamentos robustos e flexíveis em um ambiente dinâmico e complexo, e precisam responder de maneira apropriada às situações do ambiente.

Um exemplo de robô capaz de exibir comportamentos sociais apropriados é apresentado em (Arkin et al., 2001) (Arkin et al., 2003). Neste trabalho é proposta uma metodologia para o desenvolvimento de um sistema comportamental baseado em trabalhos da psicologia e da etiologia. Um modelo etiológico específico é criado com base nos estudos sobre cachorros domésticos. A metodologia emprega uma abordagem *botton-up* e *top-down* para o desenvolvimento do sistema comportamental. O sistema utiliza um processo motivacional para auxiliar na organização e na manutenção da seqüência de expressões do robô. O sistema utiliza também um conjunto de estruturas motoras-perceptuais para relacionar estímulos e respostas. Esse sistema emprega quatro níveis hierárquicos: (1) *sistemas* - uma coleção de estados motivacionais com uma certa independência entre esses estados; (2) *subsistemas* - estratégias coerentes que servem como funções gerais do sistema e quando são ativas sensibilizam o robô a certos estímulos e respostas; (3) *modos* - sub-estados motivacionais relacionados a organizações seqüenciais temporais com respeito a um estímulo; (4) *módulos motores-perceptuais* - módulos de respostas à estímulos individuais. Na Figura 2.4 é ilustrada a arquitetura deste sistema comportamental.

2.3.5 Expressão Facial

Para um robô sociável estabelecer e manter relações com humanos de maneira natural é necessário que o robô entenda as pessoas, assim como é necessário que as pessoas entendam o robô. Especificamente, o comportamento do robô e a maneira deste se expressar (expressões faciais, troca de olhar e postura, gestos, ações, etc.) deve ser compatível com a maneira como os humanos interpretam os sinais sociais de outras criaturas. Estes sinais devem permitir que uma pessoa determine as necessidades e estado interno do robô.

Recentemente, as expressões faciais têm sido exploradas nas pesquisas e projetos de robôs sociáveis, para tornar a interação com seres humanos mais natural (Ekman, 1999) (Bruce et al., 2001) (Breazeal, 2002) (Gockley et al., 2006). Uma das faces robóticas mais realísticas foi desenvolvida na Universidade de Tokyo (Kobayashi et al., 1994). Essa face foi projetada para ser semelhante a uma face humana e incorpora cabelos, dentes e uma pele de silicone. Diversos atuadores produzem movimentos faciais e expressões semelhantes às humanas. Esse robô pode ser visto na Figura 2.5.

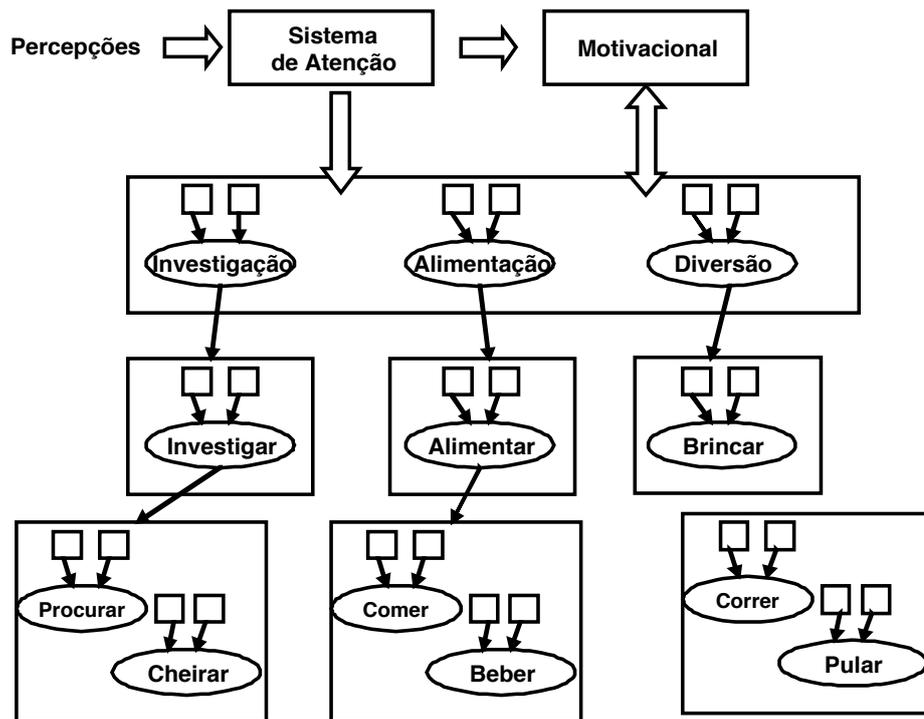


Figura 2.4: Arquitetura de um sistema comportamental. Um conjunto de estímulos é codificado e transmitido ao sistema motivacional e ao sistema comportamental do robô. O sistema motivacional calcula os valores de ativação das unidades de processamento e atualiza o estado interno do robô. As unidades de ativação competem entre si para determinar qual será a dominante que terá o direito de disparar seu comportamento correspondente. Este valor de ativação, juntamente com o conjunto de estímulos, é propagado pelos três níveis do sistema comportamental que seleciona sucessivamente o comportamento a ser emitido pelo robô. Neste modelo, o sistema motivacional também é influenciado pelos comportamentos ativos (Arkin et al., 2003).

Em (Breazeal, 2000) (Breazeal, 2002) (Breazeal, 2003a), é apresentado o robô *Kismet*. Este robô possui um sistema emocional que influencia suas expressões faciais. Desta forma, os humanos podem interpretar os sentimentos e necessidades do robô para interagir de maneira mais natural. A face deste robô possui quinze atuadores, muitos dos quais trabalham em conjunto para exibir emoções específicas como felicidade, angústia, raiva e medo. As expressões faciais de *Kismet* são geradas por meio de uma técnica baseada em interpolação. Esta interpolação é realizada empregando três dimensões correspondentes a estimulação, valência, e posição. Estes mesmos três atributos são empregados para a determinação do estado emocional e motivacional do robô a partir de fatores ambientais e internos que contribuem para o seu estado afetivo. O estado afetivo corrente ocupa um único ponto localizado neste espaço de três dimensões. Com as mudanças no estado emocional do robô, este ponto se altera dentro deste espaço. Algumas expressões faciais do robô



Figura 2.5: Expressões faciais do robô Saya (Kobayashi et al., 1994).

Kismet podem ser vistas na Figura 2.6.



Figura 2.6: Expressões faciais do robô Kismet (Breazeal, 2002).

Alternativamente ao uso de atuadores mecânicos, outra abordagem de desenvolvimento de faces com expressões é baseado na criação de modelos computacionais gráficos e técnicas de animação. Em (Bruce et al., 2001) é apresentado o robô Vikia, que possui uma face feminina renderizada em 3D. Por ser uma face graficamente construída e renderizada, ela possui muitos graus de liberdade para gerar um número de expressões que poderiam ser limitadas pelo uso de atuadores mecânicos. Foram implementadas expressões faciais para algumas emoções básicas como felicidade, tristeza, raiva, orgulho e vergonha. Tais expressões foram criadas em três níveis de intensidade para cada

emoção, e são utilizadas para acrescentar estímulos visuais emocionais à fala de Vikia. A fala do robô, a animação da cabeça e face do robô são controladas por uma linguagem de *script* que permite o sequenciamento de movimentos de cabeça, expressões faciais e fala. Esta linguagem permite a representação dos comportamentos como máquinas de estado nas quais as transições são ativas pelo módulo de percepção de estímulos do ambiente. Esse robô pode ser visto na Figura 2.7.



Figura 2.7: A face da robô Vikia (Bruce et al., 2001).

Em (Gockley et al., 2006) é apresentado um robô recepcionista que também possui um subconjunto básico de emoções sugeridas por Ekman (Ekman, 1999) e que também explora o uso de expressões faciais em uma face gráfica renderizada em 3D. O robô possui uma base mecânica com um monitor LCD que exibe a sua face e pode ser girado sobre o seu eixo vertical. Este robô pode exibir diversas expressões emocionais facilmente reconhecíveis. Ele possui um sistema emocional capaz de produzir emoções como alegria, tristeza, frustração e raiva. Para cada emoção, são definidas expressões de diferentes intensidades que podem ser exibidas pela face do robô, de acordo com a intensidade da ativação da emoção. As emoções são exibidas imediatamente após um acontecimento, permanecendo durante uma resposta verbal do robô. Esse robô pode ser visto na Figura 2.8.

2.3.6 Interação Social

Se um robô é parte do mundo físico e compartilha o ambiente dos seres humanos, é provável que este se encontre diariamente com estes, tanto em interações nas quais uma pessoa aborda o robô para solicitar uma tarefa, quanto em interações nas quais o robô procura por uma pessoa para levar alguma informação. Neste contexto, o desenvolvimento de mecanismos que permitam estabelecer uma longa interação com os seres humanos traz um importante desafio para o projeto de robôs sociáveis (Breazeal, 2004) (Gockley et al., 2005).



Figura 2.8: Expressão facial do robô Valerie (Gockley et al., 2006).

A qualidade da interação humano-robô possui um importante papel no aumento do uso dos robôs na vida diária das pessoas. Por meio da interação social efetiva, robôs serão capazes de executar muitas tarefas na sociedade humana. Estas tarefas podem incluir a execução de diversos deveres de casa, o provimento de cuidados médicos para pessoas idosas, o auxílio a pessoas com déficits motores ou cognitivos, entretenimento, educação, fornecimento de informações em lugares públicos, entre outros. Para estas e outras aplicações, faz-se necessário o desenvolvimento de robôs sociáveis que possam se comportar como parceiros dos seres humanos (Breazeal, 2002) (Salichs et al., 2006).

Na literatura foram propostos muitos mecanismos de controle para interações sociais entre robôs e seres humanos (Kanda et al., 2004) (Gockley et al., 2005).

Em (Kanda et al., 2004), é apresentado um robô capaz de identificar pessoas e de se comunicar com elas e interagir em atividades sociais, estabelecendo longas interações com humanos. Este robô interativo é capaz de atrair as pessoas para interagir e reconhecer relações amigáveis entre humanos, identificando cada pessoa de um grupo social. O robô também é capaz de exibir habilidades da atenção compartilhada, rastreando o olhar das pessoas e selecionando objetos que são o foco de atenção delas. Este robô ainda é capaz de apontar para objetos e estabelecer contato visual com seu parceiro de interação. Adicionalmente, ele pode verbalizar até 300 sentenças e reconhecer aproximadamente 50 palavras. A arquitetura do robô foi implementada como módulos de ambientação e interação que processam informações dependentes do contexto para interpretar as reações humanas. Os módulos

de ambientação e interação são compostos por um sistema de pré-condições, por um sistema de indicação que contém unidades básicas de comunicação e ação e um sistema de reconhecimento que é responsável por identificar as reações dos seres humanos. Durante uma interação, o sistema checa as pré-condições para a ativação de um módulo de ambientação e executa as ações deste módulo, se as pré-condições forem satisfeitas. Então, o sistema verifica as reações das pessoas com as quais o robô está interagindo e, conforme estas reações, efetua a transição para um outro módulo de ambientação. Estas transições são direcionadas por uma base de regras de episódios de interação. Estas regras também permitem uma transição reativa entre dois módulos de ambientação. Se o contexto do ambiente é alterado abruptamente, o conjunto de regras de episódios leva à ativação de uma transição em resposta a mudança do ambiente.

Em (Gockley et al., 2005), é apresentado um robô recepcionista projetado para realizar longas interações sociais com seres humanos. O robô é capaz de executar algumas das tarefas que uma recepcionista executaria, como fornecer os números dos escritórios das pessoas e indicar direções a serem seguidas para se chegar a um determinado escritório. Para tornar o robô uma presença natural, ele recebeu algumas características humanas como um nome, uma personalidade, um história prévia de vida e diversos enredos que são encenados com o passar do tempo. O robô possui uma estrutura de interação que inclui sua história de vida, contada em pequenos monólogos entre as interações com as pessoas. Ela também possui um sistema de reconhecimento de voz e um teclado para o recebimento de informações durante as interações. A verbalização do robô é gerada por um mecanismo do tipo *text-to-speech* e sincronizada com movimentos dos lábios e demais musculatura de face. O texto também é exibido dentro de balões próximo a sua face para auxiliar na compreensão da voz sintética. Um sensor a laser é empregado para detectar e localizar as pessoas que se movem dentro do espaço que cerca a área de trabalho do robô, provendo um campo de visão de 180 graus. A área rastreada pelo sensor é dividida em diversas regiões que são utilizadas para classificar as pessoas detectadas por estados de atenção. De acordo com estes estados de atenção, o robô pode iniciar ou finalizar uma interação, além de fornecer informações sobre a localização das determinadas áreas das instalações onde o robô está localizado.

2.3.7 *Personificação Corporal*

Seres humanos são organismos que aprendem e desenvolvem suas habilidades e conhecimentos por meio de seu corpo em interações com o ambiente e com outros organismos (Lindblom and Ziemke, 2003) (Breazeal, 2002). Pos-

suir um corpo e existir em um ambiente social é uma característica importante para um organismo (Dautenhahn, 1995).

A personificação corporal é um conceito natural para agentes biológicos, mas muito complexo de definir em agentes artificiais. Muitos pesquisadores discutem a importância da personificação corporal para se entender e gerar comportamento inteligente em modelos computacionais e robôs (Brooks, 1990) (Breazeal, 2002) (Breazeal, 2003a) (Deák et al., 2007). Por meio de um corpo, um robô pode interagir com o ambiente e interpretar os estímulos que recebe para adquirir novos conhecimentos. Adicionalmente, possuir um corpo e existir em um ambiente social pode ser muito vantajoso para um robô, assim como também para as pessoas que interagem com ele. Da perspectiva do robô, seu corpo provê um meio para experimentar e interagir com o mundo social e para interpretar suas experiências dentro deste mundo. Da perspectiva de um humano também é benéfico para o robô ter um corpo, uma vez que os humanos evoluíram para interagir socialmente com criaturas personificadas (Breazeal, 2002). Os sistemas personificados possuem a vantagem de serem capazes de se comunicar de forma pára-lingüística, emitindo sinais como gesto, expressões faciais, entonação de voz, direção de olhar e postura corporal. Estes sinais são importantes para complementar e aprimorar a mensagem do robô durante uma interação (Breazeal, 2003a). Portanto, os robôs sociáveis podem aproveitar melhor estas características humanas se eles possuírem um corpo por meio do qual eles podem interagir com o ambiente.

Um sistema é corporalmente personificado em um ambiente se existem canais de perturbação mútuas entre esse sistema e o ambiente. Quanto maior a perturbação causada por um robô em um ambiente e vice-versa, maior a integração com o ambiente e o grau de incorporação deste robô (Dautenhahn et al., 2002). Um importante benefício desta relação é que ela fornece a oportunidade para quantificar a relação do robô com o ambiente. Alguns robôs possuem maior integração que outros (Dautenhahn et al., 2002). Para um sistema corporalmente personificado interagir e exibir comportamentos no mundo real, é necessária uma coordenação apropriada entre ações e percepções. Geralmente, é aceito que ação e percepção são fortemente entrelaçadas e que o aperfeiçoamento desta junção é o resultado de um processo gradual de desenvolvimento. Um preciso controle do sistema não é possível sem percepção e vice-versa (Lungarella and Metta, 2003).

Muitos pesquisadores exploram a personificação corporal em seus projetos de robôs sociáveis (Kanda et al., 2004) (Salichs et al., 2006) (Gockley et al., 2007). Como exemplo, pode-se citar o robô Kismet apresentado em (Breazeal, 2002) (Breazeal, 2003a), capaz de interagir face-a-face com pessoas. O robô é formado por uma cabeça com aproximadamente 1.5 vezes o tamanho de

uma cabeça de uma pessoa adulta e possui um total de 21 graus de liberdade (DOF). Três DOF direcionam o olhar do robô, outros três controlam a orientação de sua cabeça, e os 15 DOF restantes controlam suas expressões faciais (pálpebras, sobrancelhas, lábios, e orelhas). Para visualizar e detectar as pessoas com quem ele interage, Kismet é equipado com um total de quatro câmeras CCD coloridas, (2 câmeras com campo estreito de visão montados em seus olhos e 2 câmeras com visão panorâmica montadas entre os olhos). Além disso, Kismet possui dois pequenos microfones (Um para cada orelha). Um terceiro microfone é usado pela pessoa que está interagindo com o robô para processar suas vocalizações. A estrutura de personificação do robô Kismet pode ser vista na Figura 2.9.

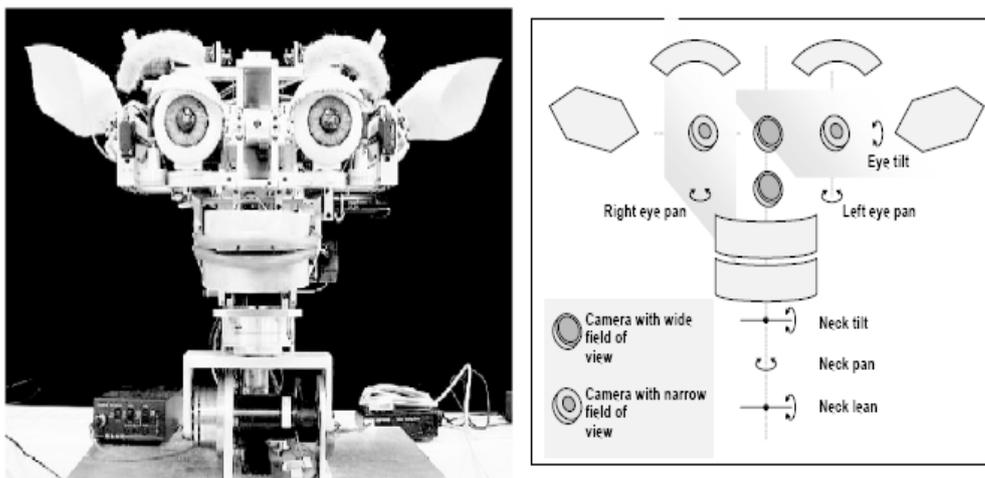


Figura 2.9: Estrutura de personificação do robô Kismet (Breazeal, 2003a).

2.4 Considerações Finais

Nesse Capítulo foram apresentadas algumas questões de projeto e considerações para o desenvolvimento de robôs sociáveis. Como pode-se verificar, robôs sociáveis podem ser desenvolvidos para diversas aplicações. Desta forma, seus projetos podem variar muito em termos de técnicas, funcionalidades e objetivos sociais.

Uma diferença fundamental entre robôs sociáveis e outros tipos de robôs é a capacidade destes em interagir com seres humanos de uma maneira mais natural. Robôs sociáveis e seres humanos precisam ser capazes de coordenar suas ações de forma a tornar suas interações produtivas para a resolução de algum problema. Não é necessário o desenvolvimento de robôs socialmente sofisticados, mas sim robôs com funcionalidades que atendam as necessidades humanas, que realizem as tarefas necessárias e que sejam naturalmente

compreendidos.

Embora muito progresso tenha sido alcançado, ainda existem muitas questões e desafios que demandam pesquisas mais aprofundadas para a exploração e o desenvolvimento de mecanismos e estruturas que permitam a utilização destes robôs na vida diária das pessoas. Com o progresso destas pesquisas sobre robôs sociáveis e o potencial que esses robôs oferecem, é esperado o aumento da utilização destes para auxiliar os seres humanos em um número de tarefas. Adicionalmente, é esperado que esses robôs trabalhem juntamente com as pessoas e as auxiliem a cumprir suas tarefas diárias.

A chave da aceitação dos robôs sociáveis é sua capacidade de interagir com os seres humanos de maneira natural. Portanto, são necessários importantes avanços no desenvolvimento de mecanismos que possibilitem interações mais naturais, bem como mecanismos que regulem o relacionamento entre robôs e humanos.

Um outro aspecto importante desta área de pesquisa é o desenvolvimento de métodos e estudos sobre os impactos da inserção destes robôs na sociedade, especialmente sobre a evolução do comportamento humano em interações humano-robô no decorrer do tempo.

Neste contexto, os processos de aprendizado e modelagem do comportamento humano evidenciados na Análise do Comportamento, uma importante teoria da Psicologia, pode levar ao desenvolvimento de métodos e estruturas promissores para a construção de robôs sociáveis capazes de aprender pela interação com o ambiente e capazes de exibir comportamentos apropriados. No próximo Capítulo são introduzidos alguns dos principais conceitos da teoria da Análise do comportamento, a teoria base deste trabalho.

Análise do Comportamento

3.1 Considerações Iniciais

A Análise do Comportamento é uma ciência, inserida na área de conhecimento da Psicologia, que se caracteriza pela análise sistemática do comportamento humano em interação com o ambiente e a aprendizagem decorrente desta interação. Neste trabalho, especificamente, o conceito primordial que embasou o desenvolvimento da arquitetura proposta é o *Comportamento Operante*.

A descrição desse tipo de comportamento envolve duas relações: a relação entre a resposta e sua conseqüência, e a relação entre a resposta e o estímulo antecedente que estava presente quando a resposta foi reforçada. Estas relações são características de todo o comportamento operante e formam a unidade básica de análise do comportamento, composta por três termos, denominada contingência tríplice: um estímulo discriminativo anterior (S^D), a resposta do agente (R) e o estímulo reforçador (S^R). A relação básica pode ser vista abaixo (Skinner, 1953) (Skinner, 1974) (Catania, 1999) (Serio et al., 2004).

$$S^D \rightarrow R \rightarrow S^R$$

Na descrição do comportamento operante, é impossível falar de qualquer uma dessas relações isoladamente. O que estabelece a ocasião em que uma próxima resposta, se ocorrer, será reforçada, é o estímulo reforçador, ou seja, a conseqüência da primeira contingência à qual um agente foi exposto. Assim, é necessário analisar quais são os controles de estímulos presentes na con-

tingência para afirmar sob que condições determinado comportamento deverá ocorrer. O controle de estímulos fornece informações sobre o comportamento: quais comportamentos serão efetivos e sob quais condições. Portanto, quando se deseja instalar um comportamento específico, deve-se planejar um arranjo de estímulos proporcionalmente específico para se afirmar que tal controle está em vigor.

O estudo do controle de estímulos representa uma área muito importante dentro da Análise do Comportamento. Quando um agente passa a responder diferencialmente na presença de determinados estímulos mas não na presença de outros, pode-se dizer que determinadas respostas deste agente estão sob o controle de estímulos. O estabelecimento do controle dos estímulos sobre a emissão da resposta é, por sua vez, produto de uma história específica de reforçamento¹. Uma história de reforçamento se caracteriza pela emissão sistemática de respostas se e somente se a dada ocasião original de reforçamento está em vigor. Por exemplo, uma história na qual a resposta foi seguida de reforço quando emitida na presença de determinados estímulos, mas não foi seguida de reforço quando emitida na presença de outros estímulos. Esta situação estabelece o reforçamento diferencial, isto é, o reforçamento de algumas respostas e de outras não, tendo como critérios os estímulos na presença dos quais a resposta é emitida.

Isso significa que, se a história de reforçamento diferencial for conhecida, é possível prever quando a resposta ocorrerá e, além disso, é possível aumentar a probabilidade de ocorrência de uma resposta apresentando os estímulos que a controlam. Dois processos estão envolvidos no estabelecimento das ocasiões em que as respostas deverão ocorrer: Discriminação e Generalização. Chama-se de Discriminação o controle de estímulos estabelecido quando, após uma história de reforçamento, alguns estímulos aumentam (S^D ou S^+) ou diminuem (S^Δ ou S^-) a probabilidade de uma resposta ocorrer em determinada situação. Estes estímulos são denominados *discriminativos* e sinalizam ocasiões em que a resposta terá conseqüências específicas (Skinner, 1953) (Serio et al., 2004). Os estímulos S^D e S^Δ podem ser apresentados sucessivamente (discriminação sucessiva) ou simultaneamente (discriminação simultânea). Após o estabelecimento da discriminação de alguns estímulos, pode-se notar que outros aspectos destes estímulos ou do ambiente (tempo, localização, brilho, cor, intensidade, tamanho) podem passar a controlar mesmas respostas. Assim,

¹do inglês *reinforcement*. Por não haver uma palavra que descreva consistentemente o processo na Língua Portuguesa, o uso do termo *reforçamento* é sancionado e amplamente utilizado na área da Análise do Comportamento. Por definição, reforçamento é um processo que se relaciona ao aumento da probabilidade de ocorrência ou emissão de respostas. Diz-se que houve reforçamento se há aumento do número de respostas claramente ocorrendo em função de uma conseqüência reforçadora apresentada de forma contingente à emissão de tal resposta

ocorre outro processo básico descrito no âmbito do controle de estímulos, denominado generalização. Uma vez que se coloca o comportamento sob controle de um estímulo, freqüentemente verifica-se que outros estímulos, que possuem propriedades semelhantes, (características físicas, tamanho, cor, brilho, intensidade) também passam a controlar o comportamento. Desta maneira, um agente pode emitir uma resposta para estímulos semelhantes que apresentem ao menos uma propriedade em comum (Serio et al., 2004). Estes processos podem ser ilustrados com o seguinte exemplo: pode-se imaginar uma criança quando começou a aprender o que era um gato. Inicialmente, tudo o que tinha um rabo e quatro patas era denominado *gato* (cachorros, cavalos). Com o passar do tempo, pessoas reforçaram diferencialmente o comportamento da criança. Se, e somente se um gato estava presente e a criança dizia *gato*, suas respostas eram reforçadas. Na presença de outros animais, como cachorros e cavalos, apenas o comportamento de dizer o nome destes outros animais era reforçado. Assim, apenas as respostas emitidas diante de determinados animais (funcionando como estímulos discriminativos) passaram a ser reforçadas. Após uma história de reforçamento, embora alguns animais tenham rabo e quatro patas, a criança aprende que, mesmo sendo animais (generalização), existem gatos, cachorros e cavalos (discriminação). Esta discriminação pode se tornar tão refinada, que a criança poderá, mais tarde, discriminar as diversas raças de gatos, como angorás, siameses e persas.

Neste Capítulo, são introduzidos alguns dos principais conceitos e processos da Análise do Comportamento, a teoria base deste trabalho de pesquisa. Este Capítulo está organizado da seguinte maneira. Na Seção 3.2, são introduzidos os conceitos sobre comportamento operante. Na Seção 3.3, é discutida a modelagem do comportamento operante. Na Seção 3.4, é apresentado o processo de discriminação operante. Na Seção 3.5, é discutida a importância da motivação para a modelagem do comportamento operante. Finalmente, na Seção 3.6, são feitas as considerações finais deste Capítulo.

3.2 Comportamento Operante

Comportamento é uma característica primária dos seres vivos. Alguns comportamentos fazem sentido em termos dos eventos que os precedem, outros comportamentos fazem sentido mais em termos dos eventos que os sucedem, ou conseqüências. Os comportamentos estabelecidos e direcionados por suas conseqüências são denominados de comportamento *operante*. A palavra operante refere-se a uma propriedade essencial do comportamento orientado por metas: eles afetam o ambiente. As conseqüências de um comportamento podem realimentar o organismo, alterando a probabilidade do comportamento

que os produziu acontecer novamente. Tais comportamentos surgem na maioria dos problemas práticos em termos de comportamento humano e também são de interesse teórico particular devido a suas características especiais (Skinner, 1953) (Staddon, 1983).

Um efeito mais específico de uma consequência foi inicialmente estudado por Edward L. Thorndike (Thorndike, 1898) apud (Catania, 1999) em um experimento bastante conhecido. Um gato trancado em uma caixa tentava escapar e, eventualmente, movia a trava que abria a porta. Quando trancado repetidamente nessa caixa, o gato gradualmente diminuía a probabilidade de emitir comportamentos aleatórios e ineficazes e, progressivamente, emitia uma resposta bem-sucedida mais e mais rapidamente (Catania, 1999).

Outro conceito importante para a definição do comportamento operante é o de *resposta*. Pode-se referir a qualquer movimento de um organismo como uma resposta, significando as ações de um organismo como resposta a eventos antecedentes. Uma resposta nunca é igual a outras respostas emitidas pelo mesmo organismo. Duas respostas, mesmo que semelhantes, podem diferir em algumas de suas propriedades, como a topografia ou o tempo. Portanto, a unidade mínima utilizada pela Análise do Comportamento para prever e controlar comportamentos não é uma resposta, mas uma classe destas. Uma classe de respostas é formada por diversas respostas semelhantes que variam em algum grau mas que levam às mesmas consequências como resultado de sua emissão. A palavra operante é usada para descrever esta classe de respostas. O termo enfatiza o fato que comportamento opera sobre o ambiente para gerar as consequências (Skinner, 1953) (Catania, 1999).

Como exemplo, pode-se citar um experimento com um pombo (Skinner, 1953). Uma instância simples na qual o pombo levanta sua cabeça é uma resposta. O comportamento denominado *levantar a cabeça* é um operante. Ele pode ser descrito não como uma ação realizada, mas como um conjunto de ações definidas pela propriedade da altura para a qual a cabeça é elevada. Neste sentido, um operante é definido por um efeito que pode ser especificado em condições físicas; o *deslocamento* a uma certa altura é uma propriedade de comportamento.

Um operante pode ser modelado por meio de um processo denominado *condicionamento operante*. No condicionamento operante, deve-se *fortalecer* um operante no sentido de tornar a resposta mais provável ou mais freqüente.

Em um experimento típico (Catania, 1999) (Serio et al., 2004), um rato é colocado em uma caixa após ter sido privado de alimento. A caixa contém uma barra na parede e, próximo à barra, encontra-se um comedouro para disponibilizar pelotas de alimento, depositadas por um mecanismo do outro lado da parede, e um ruído característico acompanha cada apresentação da

pelota. Uma lâmpada piloto ilumina a caixa, e um ruído em um alto-falante pode mascarar os sons de fora da caixa. Como primeiro passo, pode-se modelar o comportamento do rato de ir ao comedouro, reforçando tais ações com a produção de uma pelota de alimento. Estas pelotas são depositadas no comedouro e, cedo ou tarde, o rato as encontra e as come. Uma vez que isso ocorra, as pelotas continuam caindo até que, após cada apresentação, o rato se aproxime rapidamente do comedouro a partir de qualquer ponto da caixa. Uma vez completado o treino ao comedouro, pode-se modelar o comportamento do rato apontar a cabeça na direção da barra, apresentando o alimento quando ele emitir esta resposta, da mesma maneira que o treino ao comedouro. Então, pode-se alterar o procedimento de maneira que a apresentação do alimento passe a depender das pressões à barra. Eventualmente o rato pressiona a barra, a pressão produz a queda da pelota e a pelota ocasiona o comer. O rato então, provavelmente, voltará à barra e a pressionará novamente.

As pressões à barra por um rato privado de água tornam-se mais prováveis quando resultam na apresentação de água do que quando não produzem esta consequência. Da mesma maneira, as bicadas no disco tornam-se mais prováveis quando um pombo privado de alimento bica e produz comida do que quando não produz. Neste mesmo contexto, quando uma criança faz um escândalo, em um supermercado por exemplo, pedindo alguma coisa que ela queira e sua mãe a atende, é provável que aumente a frequência da criança emitir este comportamento, pois ela receberá novamente a atenção de sua mãe. Estes processos ilustram o princípio do reforço: o responder aumenta quando favorece o acesso a reforçadores (Catania, 1999).

A terminologia do reforço inclui o termo *reforçador* como estímulo e os termos *reforçar* e *reforço* ou *reforçamento*, tanto como operação quanto como processo. Por exemplo, quando as pressões à barra por um rato resultam no acesso a pelotas de alimento, pode-se dizer que as pelotas de alimento são reforçadoras ou que as pressões à barra são reforçadas com pelotas de alimento. Embora um reforçador seja uma espécie de estímulo, o reforço não é nem um estímulo e nem uma resposta. Como uma operação, o reforçamento é a apresentação de um reforçador quando uma resposta ocorre. Portanto, pode-se dizer que a operação de reforçamento é efetuada sobre as respostas e não sobre os organismos (Catania, 1999).

No caso do experimento com o pombo, a comida é um reforçador e a apresentação da comida, quando uma resposta é emitida, é o reforço ou reforçamento. Ainda, o operante é definido pela propriedade na qual o reforço é contingente à emissão da resposta, neste caso, a altura para a qual a cabeça deve ser elevada. A mudança na frequência com que a cabeça é erguida a esta altura é denominado processo de condicionamento operante (Skinner, 1953)

(Catania, 1999). Nesta condição é verificada uma diminuição da ocorrência da resposta, mas tal diminuição é apenas temporária. Assim, diz-se que a punição apenas tem o efeito temporário sobre o comportamento.

No processo de reforçamento, um organismo pode repetir uma resposta pois, para ele, as conseqüências são *agradáveis* ou *satisfatórias*. O reforço também pode ser efetivo porque reduz um estado de privação. Eventos que são denominados reforçadores são de dois tipos. Alguns reforços consistem em apresentar estímulos que acrescentam algo de valor ao indivíduo, como por exemplo, comida, água, ou contato sexual. Estes podem ser denominados reforçadores positivos. Outros consistem em remover estímulos aversivos, que constituem uma condição antecedente aversiva, como por exemplo, um barulho alto, frio ou calor intensos, uma luz extremamente brilhante, ou um choque elétrico. Estes podem ser denominados reforçadores negativos. Em ambos os casos o efeito de reforço é o mesmo: a probabilidade da emissão de uma resposta é aumentada. Então, o reforçamento positivo é caracterizado pela apresentação de estímulos *agradáveis* e o reforçamento negativo é caracterizado pela remoção de estímulos *aversivos* (Skinner, 1953) (Catania, 1999).

Existe uma outra relação, a *punição*, em que as conseqüências do responder o tornam menos provável. Um estímulo que reforça uma resposta quando é produzido como conseqüência desta, pode ter uma função diferente quando é removido por outra resposta. A remoção de um reforçador pode ter a conotação de punição. Inversamente, o estímulo que pune a resposta que o produz, pode reforçar a resposta que o elimina. Por exemplo, o dinheiro pode reforçar, como quando uma criança é paga por concluir uma tarefa, no entanto, sua remoção pode punir, quando a mesada dessa criança é retirada devido a um comportamento inadequado (Catania, 1999).

Assim como o reforço, o termo punição é empregado com referência a operações e processos. Desta forma, afirmar que uma resposta foi punida pode significar que a resposta produziu um estímulo punitivo ou que houve um decréscimo no responder devido à produção deste estímulo. Da mesma maneira que o reforço, é mais indicado restringir o termo punição à terminologia de operações e descrever diretamente o processo em termos das alterações no responder. O efeito da punição é simplesmente o oposto do efeito do reforço. O reforço e a punição são simétricos: o primeiro aumenta o responder, enquanto a última o diminui, mas seus efeitos continuam enquanto os procedimentos são mantidos e podem desaparecer depois que eles são interrompidos, quando então o responder do organismo retorna aos níveis anteriores ao início dos procedimentos (Catania, 1999).

Um outro processo relacionado ao reforçamento é denominado de *extinção*

operante. Este processo ocorre quando um reforçamento é interrompido, tornando uma resposta menos e menos freqüente. Por exemplo (Skinner, 1953), se a comida é retida durante um experimento com um pombo este deixará, eventualmente, de erguer a sua cabeça. Se a mesma criança do exemplo do supermercado faz um escândalo, mas a sua mãe não a atende ou a ignora, é provável que a freqüência da emissão do comportamento inadequado pela criança diminua.

A ausência de um reforçamento em uma resposta previamente reforçada leva não apenas a extinção operante, mas também a uma reação que pode ser classificada como frustração ou raiva. Por exemplo, após parar de receber comida ao pressionar a barra, um rato pode aumentar a velocidade e o número de pressões à barra. O comportamento durante a extinção é o resultado do condicionamento que o precedeu. Se foram reforçadas só alguns respostas, a extinção acontece rapidamente. Uma história longa de reforço é seguida por uma extinção lenta. Não se pode predizer a resistência para extinção da probabilidade de resposta observada em determinado momento sem saber a história de reforçamento prévia da resposta alvo (Catania, 1999).

3.3 Modelagem do Comportamento Operante

Em referência aos exemplos de resposta de um rato ao pressionar uma barra, ou de um pombo ao bicar um disco, se simplesmente se colocar um rato diante de uma barra ou um pombo diante de um disco, corre-se o risco de não se observar as respostas de pressionar ou bicar. O reforço pode não ter efeito se a resposta a ser reforçada nunca for emitida. Ao invés de se esperar pela resposta desejada, o experimentador pode gerá-la por meio de aproximações sucessivas de outras respostas que sejam similares ou próximas à desejada, por meio do *condicionamento operante* ou *modelagem* (Skinner, 1953) (Catania, 1999).

Por meio deste processo, os comportamentos operantes podem ser reforçados, em aproximações sucessivas, até a emissão do comportamento desejado. Por exemplo, para fazer um pombo bicar um disco, pode-se proceder da seguinte maneira: pode-se dar comida ao pombo quando ele virar ligeiramente sua cabeça na direção do disco, a partir de qualquer parte da gaiola, aumentando a freqüência de tal comportamento. Então, deve-se manter este reforço até que um movimento leve seja feito em direção ao disco. Pode-se então continuar reforçando posições sucessivamente mais próximas ao disco, reforçando então somente quando a cabeça apresenta movimentos na direção do mesmo e, finalmente, só quando o pombo bicá-lo (Skinner, 1953) (Catania, 1999).

A modelagem é baseada no *reforço diferencial*, isto é, em estágios sucessi-

vos, algumas respostas são reforçadas e outras não. Além disso, à medida que o responder se altera, os critérios para o reforço diferencial também mudam, em aproximações sucessivas da resposta a ser modelada. A propriedade do comportamento que torna a modelagem efetiva é a variabilidade do comportamento. Duas respostas nunca são uma mesma resposta e o reforço de uma resposta produz um espectro de respostas diferentes da resposta reforçada ao longo de algumas dimensões como topografia (forma), força, magnitude e direção. Dessas respostas, algumas estarão mais próximas da resposta a ser modelada do que outras e podem, então, ser selecionadas, para serem reforçadas em seguida. Reforçar estas respostas, por sua vez, será seguido de outras mais, algumas das quais podem estar ainda mais próximas da resposta a ser modelada. Assim, o reforço poderá ser usado para mudar o espectro de respostas, até que a resposta a ser modelada ocorra (Catania, 1999).

Como exemplo de reforço diferencial, considere um rato privado de alimento em um caixa com uma fenda horizontal de 30 centímetros de comprimento em uma das paredes (Antonitis, 1951) apud (Catania, 1999). Nesta caixa, células fotoelétricas registram o local, ao longo da fenda, no qual rato introduz seu nariz. A fenda é dividida em segmentos de 2 centímetros e cada segmento é rotulado de 1 a 15, da esquerda para a direita. Existe também um mecanismo que pode apresentar pelotas de alimento em um comedouro na parede oposta à da fenda e as apresentações de pelotas são acompanhadas por um ruído característico, quando então o rato rapidamente vai ao comedouro e come sempre que uma pelota é apresentada. Inicialmente, o rato passa apenas um tempo curto próximo à fenda na parede e, ocasionalmente, cheira a fenda e introduz o focinho nela, ao mesmo tempo que se movimenta ao longo da parede, mas essas respostas são relativamente ocasionais e não possuem qualquer relação sistemática com as posições ao longo da fenda apresentando uma baixa distribuição das respostas de cheirar a fenda. Então, pode-se passar a reforçar somente as respostas de introduzir o focinho na fenda, nas posições entre 9 e 12. A partir deste momento, ocorre uma alteração na distribuição das respostas de cheirar a fenda, com um aumento das respostas de introduzir o focinho nas posições 9 a 12, mas também verifica-se que o reforço não aumenta somente o responder nas posições relacionadas com o reforçamento. Ele aumenta a frequência das respostas em todas as posições da fenda, reforçando outras respostas próximas às respostas desejadas. A medida que se continua a reforçar somente as posições da fenda entre 9 e 12, percebe-se um aumento na frequência da emissão das respostas relacionadas a estas posições e uma diminuição na frequência da emissão das respostas relacionadas às demais posições, até um limite no qual as distribuições não mais se alteram (Catania, 1999). Esta alteração do comportamento do rato

pode ser visto na Figura 3.1.

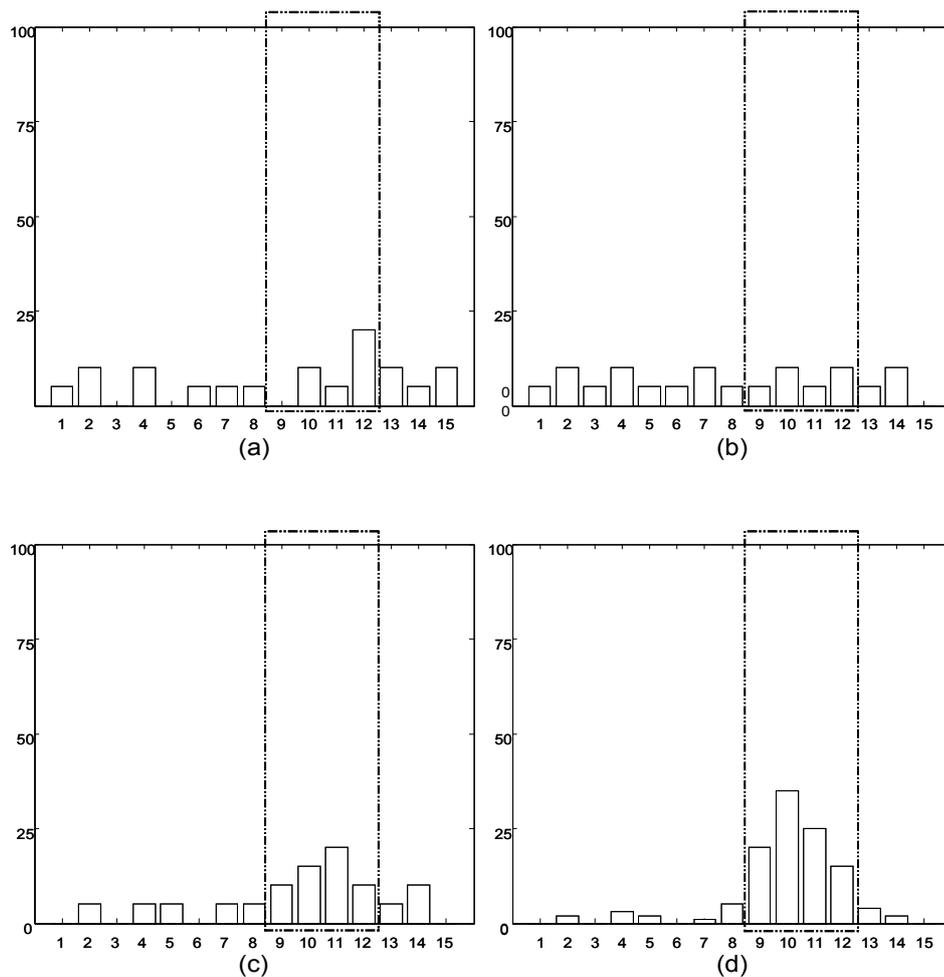


Figura 3.1: Distribuições hipotéticas de respostas. As distribuições constantes nos gráficos ilustram os efeitos do reforço diferencial sobre a localização da resposta na fenda da caixa. O eixo x dos gráficos mostram as posições de uma fenda com segmentos rotulados de 1 a 15 e a caixa pontilhada delimita as posições da fenda que estão sendo reforçadas. Inicialmente, o rato ocasionalmente cheira a fenda e introduz o focinho nela, ao mesmo tempo que se movimenta ao longo da parede, emitindo respostas relativamente ocasionais e que não possuem qualquer relação sistemática com as posições ao longo da fenda, apresentando uma baixa distribuição das respostas de cheirar a fenda (gráfico a). Então, pode-se passar a reforçar somente as respostas de introduzir o focinho na fenda, nas posições entre 9 e 12 (gráfico b). A partir deste momento, ocorre uma alteração na distribuição das respostas de cheirar a fenda, com um aumento das respostas de introduzir o focinho nas posições 9 a 12 (gráfico c). À medida que se continua a reforçar somente as posições da fenda entre 9 e 12, percebe-se um aumento na frequência da emissão das respostas relacionadas a estas posições e uma diminuição na frequência da emissão das outras respostas, até um limite no qual as distribuições não mais se alteram (gráfico d). Adaptado de (Catania, 1999).

Desta maneira, pode-se construir operantes complexos que nunca apareceriam no repertório do organismo de maneira natural. Pelo reforço de séries

de aproximações sucessivas, pode-se instalar uma nova e complexa resposta que passa então a ser emitida a uma probabilidade muito alta. Por exemplo, o ato total de virar em direção ao disco, a partir de qualquer ponto a caixa, caminhar em direção a ele, levantar a cabeça e bicá-lo, pode parecer ser uma unidade funcionalmente coerente de comportamento; mas é construído por um processo ininterrupto de reforço diferencial de comportamento a partir de diversas respostas existentes ou implantadas no repertório do pombo (Skinner, 1953).

Quando se inspeciona um comportamento complexo, assim como o do pombo bicar o disco, pode ser conveniente distinguir as diversas respostas que compõem este comportamento e que se diferem uma da outra pela topografia e pelas conseqüências que produzem (Skinner, 1953). Quando se *quebra* uma seqüência de comportamento em seus componentes, pode-se tratar esta seqüência como uma sucessão de operantes diferentes, cada um definido pela conseqüência reforçadora de produzir uma oportunidade de emitir o próximo, até que a seqüência seja terminada por um último reforçador. Esse tipo de seqüência é denominado uma cadeia de respostas. Um exemplo de sequenciamento de respostas é o do bicar um disco por um pombo, no qual este deve se orientar na direção do disco, produzindo a ocasião para que ele se aproxime e produzindo a ocasião para ele dê uma bicada no mesmo. Qualquer segmento da seqüência serve à dupla função de reforçar a última resposta e de produzir as condições que ocasionam a resposta seguinte. Estes estímulos, ou ocasiões, discriminam o contexto para a emissão da próxima resposta da seqüência e exercem a função de reforçador condicional. Um reforçador condicional é um estímulo que se torna reforçador por ser associado com outro estímulo reforçador. Neste exemplo, a resposta do pombo orientar-se na direção do disco foi inicialmente reforçada com alimento e, portanto, o estímulo visual do disco tornou-se um reforçador condicional (Catania, 1999). Desta maneira, um comportamento pode ser *quebrado* em partes para facilitar a análise do mesmo. Estas partes são os *atos* pelos quais, no vocabulário dos leigos, o comportamento é dividido (Skinner, 1953).

Durante um processo de modelagem, embora o reforço de um operante seja sempre uma questão de selecionar a magnitude de certas resposta desejadas, pode-se distinguir entre produzir uma nova unidade de resposta relativamente completa e produzir mudanças leves em uma unidade de resposta existente, tornando-a mais efetiva para o comportamento desejado. No primeiro caso, o interesse é no comportamento adquirido. No segundo, o interesse é em como o comportamento é refinado. A contingência que aprimora uma habilidade é denominada de *reforço diferencial de respostas* que possuem propriedades especiais. Um exemplo deste processo é o do rato privado de alimento em uma

caixa com uma fenda horizontal, apresentado anteriormente. Em relação ao primeiro caso, o reforço que desenvolve novas habilidades deve ser imediato. Caso contrário, a precisão do efeito diferencial será perdida. Na prática de rifle (Skinner, 1953), por exemplo, propriedades das respostas, de escala extremamente pequenas, são diferencialmente reforçadas por um tiro certo ou um tiro perdido. Propriedades desta magnitude somente podem ser selecionadas se o diferencial de reforço for imediato (Catania, 1999).

3.4 Discriminação Operante

Muitos comportamentos operantes adquirem conexões importantes com o ambiente. Essas conexões podem ser modeladas submetendo um indivíduo a uma contingência, denominada de contingência de três termos, por meio de uma história de reforçamento. Pode-se descrever uma contingência de três termos dizendo que um estímulo é a ocasião na qual uma resposta é seguida por um reforçamento.

Quando o responder é reforçado apenas na presença de alguns estímulos, diz-se que o reforço é correlacionado com aquele estímulo. Uma classe de respostas criada por este reforço diferencial em relação às propriedades do estímulo é denominada de *operante discriminado* e o processo pelo qual isto ocorre é chamado *discriminação*. Sua importância em uma análise teórica, como também no controle prático do comportamento é óbvia: quando uma discriminação foi estabelecida, pode-se alterar a probabilidade de uma resposta apresentando ou removendo o estímulo discriminativo (Skinner, 1953) (Skinner, 1974) (Staddon, 1983) (Catania, 1999).

É bem provável que não exista uma classe operante sem estímulos discriminativos. As bicadas de um pombo não podem ser emitidas na ausência de um disco, e as pressões à barra por um rato não podem ser emitidas na ausência de uma barra. As características que permanecem relativamente constantes ao longo de um experimento, como a própria câmara e os demais dispositivos que ela contém, às vezes, são referidos como *estímulos contextuais*. Os estímulos discriminativos correspondem aos estímulos coloquialmente denominados de sinais ou pistas. Estes estímulos estabelecem a ocasião em que as respostas têm conseqüências e pode-se dizer que eles ocasionam as respostas. Por exemplo, em um experimento no qual os ratos pressionam uma barra na presença e na ausência de luz programada alternadamente. Quando a luz está acesa, as pressões à barra são ocasionalmente reforçadas com alimento. Quando a luz está apagada, as pressões à barra não são reforçadas. Após uma história de reforçamento, o rato irá, eventualmente, pressionar a barra somente na presença de luz (Catania, 1999).

Após um processo de discriminação, o comportamento operante fica sob controle dos estímulos que antecedem a contingência, desde que só algumas respostas são reforçadas automaticamente pelo próprio corpo do organismo ou são reforçadas pelo ambiente. O controle de estímulos, exercido pelo corpo e pelo ambiente, possui um significado biológico óbvio. Se fosse igualmente provável que todo o comportamento acontecesse em todas as ocasiões, o resultado seria caótico. É obviamente vantajoso que uma resposta só aconteça quando for provável que esta será reforçada (Skinner, 1974).

As contingências de três termos, que produzem os operantes discriminados, podem ser de diversos tipos. Por exemplo, a visão de um objeto pode ser a ocasião para a ação (emissão da resposta) de manipulá-lo, resultando em uma consequência positiva, que se traduz na sensação tátil de segurar o objeto. Outras conexões entre as propriedades de objetos fornecem outros tipos de contingências que conduzem a mudanças semelhantes do comportamento. Por exemplo, em um pomar no qual maçãs vermelhas são doces e todas as outras são azedas, o comportamento de escolher e comer as maçãs pode ser controlado pela cor vermelha do estímulo (Skinner, 1974).

O ambiente social também pode controlar a construção de diversas contingências de três termos. Um sorriso ou uma demonstração de raiva pode sinalizar a ocasião na qual a aproximação social se dará ou não com aprovação. Desta forma, as expressões faciais caracterizam as ocasiões para a emissão de comportamentos relacionados a situações sociais. Um semáforo indicando a luz verde pode ser a ocasião na qual avançamos por um cruzamento com o carro, mas não se ele estiver vermelho (Skinner, 1974) (Catania, 1999).

Pode-se empregar a discriminação operante de dois modos. No primeiro, os estímulos que já se tornaram discriminativos podem ser manipulados para alterar as probabilidades da emissão das respostas desejadas. Os indivíduos fazem isto explicitamente e quase continuamente quando fazem críticas construtivas e elogios a um trabalho ou quando controlam o comportamento de uma criança, por exemplo. No segundo, pode-se construir e controlar uma discriminação para se ter certeza que um estímulo futuro terá um determinado efeito quando e se este aparecer. A educação, por exemplo é em grande parte uma questão de estabelecer tais repertórios discriminativos. Na educação, são empregadas diversas contingências que geram comportamentos como o de olhar para os lados antes de cruzar as ruas, agradecer as pessoas nas ocasiões apropriadas, fornecer as respostas corretas ao serem perguntadas sobre eventos históricos, operar máquinas, comprar livros, entre outros (Skinner, 1974).

3.5 Privação e Saciação

O comportamento operante dos animais superiores depende da sua motivação, além do valor da recompensa ou punição como conseqüência da emissão de uma resposta (Staddon, 1983). Por exemplo, a probabilidade de beber fica muito alta sob a privação severa de água e muito baixa sob a saciação excessiva. Desta forma, é razoável assumir que a probabilidade sempre se mantém entre estes dois extremos e que, se a privação é alterada, a probabilidade se orienta de um ponto a outro (Skinner, 1974).

Neste contexto, a *motivação* é uma palavra utilizada para se referir à razão ou razões para este aumento ou diminuição da probabilidade de se emitir uma resposta. Estas razões podem incluir necessidades básicas como comida ou um objeto desejado, passatempos, meta, objetivos e ideais. A motivação para um comportamento também pode ser atribuída a razões menos-arentes como altruísmo ou moralidade (Geen, 1994)

Um determinado ato de privação normalmente aumenta simultaneamente a força (probabilidade de emissão) de diversos tipos de comportamento. Por exemplo, quando um adulto fica por muito tempo sem água, um grupo grande de operantes é fortalecido. Ele não somente bebe a água mais prontamente, quando um copo de água é apresentado, mas ele também se ocupará de muitas outras atividades que conduzem à ingestão de água, como ir para a cozinha, pedir um copo de água, entre outras (Skinner, 1974). Pode-se dizer que a privação de água ocasiona a motivação do ser humano para a emissão de tais comportamentos.

Em condições normais um organismo bebe porque precisa de água, dá um passeio porque precisa de exercício, respira mais rapidamente e profundamente porque quer ar e se alimenta por causa da fome. Necessidades, desejos e fome são exemplos bons das causas internas ou estados internos. Necessidades e desejos são associados frequentemente com o psíquico ou o mental, enquanto a fome é associada com o fisiológico. Entretanto, estes termos são livremente utilizados para se fazer referência a privação ou saciação (Skinner, 1974).

Necessidades e desejos são termos convenientes no discurso casual. Uma necessidade, ou desejo, pode ser redefinida como uma condição que é o resultado de privação e poderiam caracterizar-se por uma probabilidade especial de resposta. Um outro termo utilizado algumas vezes é a palavra *direcionador*. Este termo é um modo conveniente de recorrer aos efeitos de privação e saciação e de outras operações que alteram a probabilidade do comportamento ocorrer da mesma maneira. Há muitos modos de mudar a probabilidade de um organismo comer. Entretanto, um único tipo de privação fortalece muitos

tipos de comportamento. O conceito de fome como um direcionador reúne estas diversas relações em um único termo (Skinner, 1974).

Privação e saciação estão relacionados ao reforço de operantes. Para um organismo faminto, a comida pode reforçar uma resposta e saciar o direcionador *fome*. Estes processos de reforçamento e saciação são distintos, mas há uma conexão inevitável entre eles: o efeito do reforçamento de um operante não será observado se o organismo não foi apropriadamente privado. O resultado do reforçamento não é somente o fortalecimento de um comportamento, mas sim a redução de um determinado estado de privação. O reforçamento coloca, então, o comportamento sob o controle de uma privação apropriada. Por exemplo, após se condicionar um pombo a esticar seu pescoço, reforçando esta resposta com comida, a variável que controla esta resposta passa a ser a privação de comida (Skinner, 1974). Então, pode-se dizer que a privação de comida ocasiona a motivação do pombo para a emissão do comportamento de esticar seu pescoço.

É possível, pelo condicionamento e extinção de uma resposta sob diferentes graus de privação, se verificar o efeito da privação em detalhes. Se um experimentador reforçar uma resposta em um grupo de organismos com o mesmo nível de privação e extingui-la em subgrupos com diferentes níveis, este poderá verificar que o número de respostas emitidas durante a extinção é uma função da privação. Por exemplo, quanto mais faminto estiver um rato condicionado anteriormente a pressionar uma barra para produzir comida, maior será o número de pressões na barra que este emitirá durante extinção. Por outro lado, se o experimentador condicionar um grupo de organismos com níveis diferentes de privação e extinguir o comportamento ao mesmo nível, ele verificará que as duas curvas de extinção contêm o mesmo número aproximado de respostas. Portanto, o efeito da privação é sentido durante a extinção e não durante o condicionamento (Skinner, 1974).

3.6 Considerações Finais

Os processos evidenciados na Análise do Comportamento e apresentados neste Capítulo denotam a maneira pela qual a aprendizagem pode ocorrer. Outros processos mais complexos ocorrem a partir destes.

No entanto, para este trabalho, estes processos básicos são importantes na proposta de construção de uma arquitetura robótica de controle capaz de simular o aprendizado de um agente por interações sociais e exibir comportamentos apropriados. A principal hipótese de pesquisa é de que o desenvolvimento de mecanismos e estruturas inspiradas na Análise do Comportamento pode levar a uma arquitetura promissora para a construção de robôs sociá-

veis capazes de aprender pela própria interação com o ambiente e capazes de exibir comportamento social apropriado.

As pesquisas bibliográficas realizadas para este trabalho, principalmente sobre a robótica sociável e sobre a teoria da Análise do Comportamento, levaram à definição do escopo deste trabalho de pesquisa e de algumas técnicas computacionais empregadas para a implementação das estruturas e mecanismos necessários para a realização do trabalho proposto. Desta forma, no próximo Capítulo, são apresentados os principais trabalhos relacionados a esta pesquisa.

Arquiteturas Robóticas

4.1 Considerações Iniciais

Uma arquitetura define um sistema em termos de componentes e as interações entre tais componentes (Shaw and Garlan, 1996). No contexto da robótica, as arquiteturas definem como são organizadas as tarefas de gerar ações a partir da percepção do ambiente (Russell and Norvig, 2003). Arquitetura robótica é uma área do conhecimento dedicada ao projeto de robôs altamente especializados por meio da integração de blocos de construção de software. Uma arquitetura é uma abstração do sistema de controle e o sistema de controle é a realização da arquitetura (Arkin, 1998).

Muitas abordagens de arquiteturas robóticas foram propostas na literatura (Brooks, 1986) (Brooks, 1991) (Arkin, 1998) (Mataric, 1999). Em termos de organização, as arquiteturas robóticas podem ser *horizontais*, nas quais as tarefas do sistema de controle são divididas em várias sub-tarefas baseadas em suas funcionalidades; *verticais*, nas quais a divisão das tarefas de controle é dividida em camadas de abstração de tarefas; *híbridas* que integram componentes verticais e horizontais.

Em termos de estratégias de controle, três abordagens principais são fortemente consideradas (Arkin, 1998) (Mataric, 1999): arquiteturas deliberativas, arquiteturas reativas e arquiteturas híbridas. Na Figura 4.1 é apresentado todo o espectro de estratégias de controles para arquiteturas robóticas. O lado esquerdo da figura representa métodos que empregam raciocínio deliberativo e o direito representa o controle reativo. Um robô que emprega raciocínio deliberativo requer conhecimento relativamente completo sobre o mundo e usa este conhecimento para prever o resultado de suas ações, uma habili-

dade que o permite aperfeiçoar suas ações relativas ao seu modelo do mundo. Entretanto, se a informação que o mecanismo de raciocínio utiliza é inexata ou desatualizada em relação a novos estados do ambiente, o resultado do processo de raciocínio pode apresentar sérios problemas. Em um mundo dinâmico, no qual objetos podem se mover arbitrariamente, é potencialmente perigoso confiar em informação prévias que podem não ser válidas. Em um outro extremo, um robô empregando raciocínio reativo não requer um modelo do mundo para executar suas tarefas. Esta estratégia integra percepção e ação, tipicamente no contexto de comportamentos, para produzir respostas rápidas em um mundo não estruturado e dinâmico.

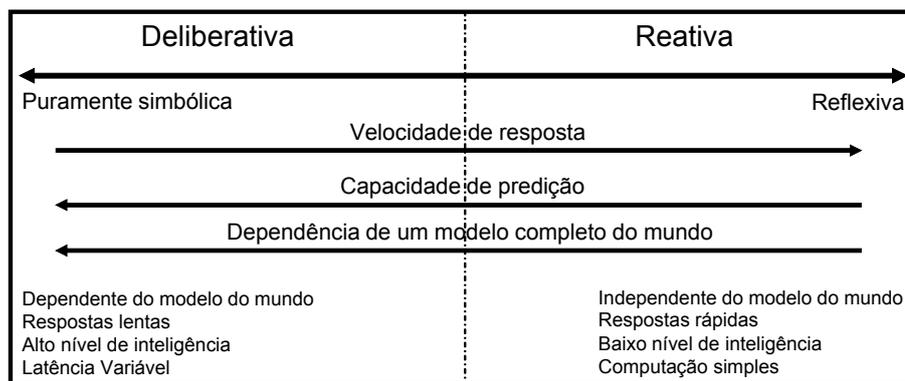


Figura 4.1: Estratégias de arquiteturas robóticas. As setas unidirecionais representam um aumento da respectiva característica indicada pela mesma. O lado esquerdo da figura representa as arquiteturas que empregam raciocínio deliberativo e o lado direito representa o controle reativo. Métodos que utilizam raciocínio deliberativo requerem um conhecimento relativamente completo do mundo e usam este conhecimento para predizer o resultado das ações. Os métodos de controle reativo não necessitam de modelos do mundo. Eles simplesmente reagem ao ambiente, fornecendo respostas imediatas mesmo para situações desconhecidas. Entre estes dois extremos, encontram-se as arquiteturas híbridas, que integram componentes deliberativos para o planejamento de longo prazo e componentes reativos para a produção de respostas rápidas a eventos e situações inesperadas (Arkin, 1998).

Neste Capítulo, são introduzidos os conceitos sobre arquiteturas robóticas e apresentadas as principais arquiteturas existentes na literatura. Este Capítulo está organizado da seguinte maneira. Nas Seções 4.2, 4.3 e 4.4, são apresentadas, respectivamente, algumas das principais arquiteturas deliberativas, reativas e híbridas propostas na literatura. Na Seção 4.5, são apresentadas algumas das principais arquiteturas de controle de robôs sociáveis existentes, que serviram como base para este trabalho, além de algumas arquiteturas propostas mais recentemente. Finalmente, na Seção 4.6, são feitas as considerações finais deste Capítulo, com uma breve discussão sobre as arquiteturas de controle de robôs sociáveis apresentadas.

4.2 Arquiteturas Deliberativas

Arquiteturas deliberativas são geralmente baseadas nos conceitos da Inteligência Artificial simbólica (Newell and Simon, 1976). Embora essa técnica tenha demonstrado altos níveis de sofisticação, também evidenciaram muitas limitações. Brooks (Brooks, 1986) se refere a esta estratégia de controle como o modelo SMPA (*Sense - Model - Plan - Act* ou Detectar - Modelar - Planejar - Agir). Estas arquiteturas empregam toda a informação sensorial disponível e todo o conhecimento interno para criar um plano de ações. Um modelo simbólico do mundo é representado explicitamente e todas as decisões são tomadas a partir do raciocínio lógico sobre este modelo. O robô primeiro recebe os dados do ambiente por meio de seus sensores e depois utiliza estes dados para construir um modelo do mundo. Então, empregando este modelo, o robô gera um plano para conseguir atingir seus objetivos e finalmente executa este plano. O controle procura por todos os possíveis planos de ação até encontrar um satisfatório para uma determinada situação. Esta seqüência de busca pode tomar muito tempo e, portanto, esta classe de arquitetura pode não ser apropriada para problemas reais nos quais os robôs precisam responder rapidamente. Além disso, existe freqüentemente um problema em se traduzir o mundo real em uma descrição simbólica precisa e adequada. Muitas arquiteturas deliberativas foram propostas na literatura (Laird et al., 1987) (Carbonell et al., 1989) (Albus et al., 1989) (Carver and Lesser, 1994). Esta Seção apresenta algumas das principais arquiteturas existentes.

A arquitetura SOAR (Laird et al., 1987) é uma arquitetura de propósito geral para auxiliar o projeto e a construção de componentes básicos de inteligência. A arquitetura SOAR emprega representação simbólica e utiliza uma base de conhecimento para a resolução de tarefas. Esta arquitetura pode não ser apropriada para problemas reais nos quais a base de conhecimento cresce rapidamente. Nesta arquitetura, existe uma estrutura simples para todas as tarefas e sub-tarefas existentes. Ela possui uma representação simples do conhecimento permanente na forma de regras de produção; uma representação simples para o conhecimento temporário, na forma de objetos que contêm atributos e valores; um mecanismo simples para a geração de metas e um mecanismo de aprendizagem. Todas as decisões nesta arquitetura são tomadas combinando o conhecimento adquirido pelos sensores, o conhecimento presente na memória de trabalho e o conhecimento da base de regras e das memórias de longo prazo. Na Figura 4.2 são ilustradas as principais estruturas desta arquitetura.

A arquitetura PRODIGY (Carbonell et al., 1989) utiliza a representação simbólica do conhecimento e foi concebida inicialmente como um sistema de In-

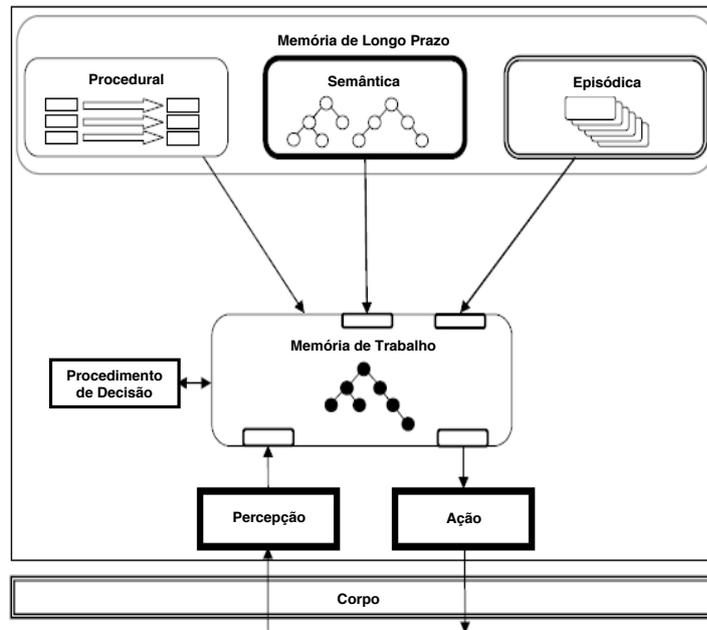


Figura 4.2: Estrutura da arquitetura SOAR. As setas representam o fluxo de informações dentro da arquitetura. Todo o processo de decisão é baseado na combinação das informações dos sensores (*Percepção*), das informações contidas na *memória de trabalho* e das informações contidas na memória permanente (*memória de longo prazo*). Estas decisões são então transmitidas para os atuadores (*Ação*) que agem sobre o *corpo* do robô para produzir as ações desejadas (Lehman and Rosenbloom, 2005).

teligência Artificial para testes, desenvolvimento de raciocínio baseado em regras de aprendizado e para a resolução de problemas de planejamento. Esta arquitetura é composta por um planejador de propósito geral, uma base de conhecimento e diversos módulos de aprendizagem projetados para reduzir o tempo de planejamento e aumentar a qualidade do conhecimento no domínio do problema. O planejamento no domínio é especificado como um grupo de operadores. Cada operador corresponde a uma ação de planejamento descrita em termos de seus efeitos e condições. Os operadores da PRODIGY podem ser organizados em diversos níveis de abstração. Na Figura 4.3 é ilustrada a organização desta arquitetura. Entre os principais componentes desta arquitetura estão:

- **EBL**: um operador de aprendizagem que adquire regras de controle a partir do solucionador de problemas. As explicações são construídas a partir das teorias que descrevem o domínio do problema e aspectos relevantes dos solucionadores de problemas da arquitetura;
- **Estático**: um método para o aprendizado de regras de controle pela análise das descrições do domínio da PRODIGY antes da resolução do problema. O Estático produz regras de controle sem utilizar qualquer

exemplo de treinamento. Ele pode ser visto como um compilador para os domínios da PRODIGY;

- **Validador:** um módulo de aprendizagem-por-experimentação para refinar o conhecimento do domínio que está incompleto ou incorretamente especificado;
- **Aprendiz de Abstração:** um módulo de aprendizagem para abstração e planejamento. O conhecimento é dividido em múltiplos níveis de abstração baseados em uma análise do domínio. Então, durante a resolução de problemas, a PRODIGY busca inicialmente, por uma solução em um espaço abstrato e então utiliza esta solução para guiar a procura por soluções em espaços de problema mais detalhados;
- **Extrator de Derivações:** um módulo de derivações de analogia que utiliza problemas semelhantes previamente solucionados para resolver novos problemas.

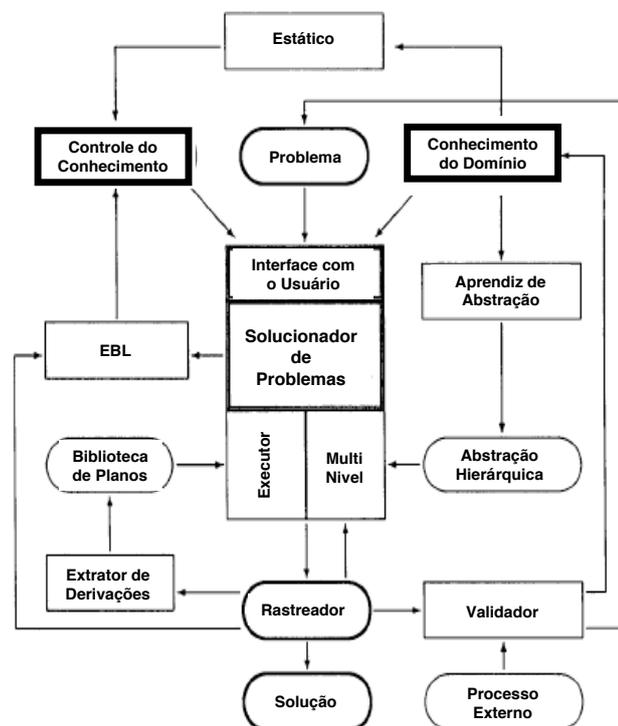


Figura 4.3: Arquitetura PRODIGY. As setas representam o fluxo de informações dentro da arquitetura. Esta arquitetura é composta por um planejador de propósito geral, uma base de conhecimento e diversos módulos de aprendizagem como o EBL, o Estático, o Aprendiz de Abstração e o Extrator de Derivações (Carbonell et al., 1989).

A arquitetura NASREM (NASA Standard Reference Model) (Albus et al., 1989) emprega uma organização hierárquica em diversos níveis. Cada nível hierárquico possui um grupo diferente de tarefas. Quanto mais alto o nível na hierarquia, mais complexa é a tarefa executada. Uma tarefa em um nível hierárquico pode empregar uma ou mais tarefas de níveis mais baixos. O primeiro nível é baseado em princípios de controle e é responsável por computar os comandos motores como o resultado de uma função de realimentação do estado do sistema. No segundo nível as dinâmicas mecânicas e grupos de comandos motores são computados. No terceiro nível são executadas as rotinas de detecção e desvio de obstáculos. No quarto nível as tarefas relacionadas a objetos são decompostas em comandos para níveis inferiores. No quinto nível as tarefas relacionadas a grupos de objetos são decompostas em seqüências de ações. No sexto nível são alocados os recursos e conhecimentos para o cumprimento das metas e objetivos do sistema. Esta arquitetura também é seqüencial e possui três seções: decomposição de tarefas, modelagem do mundo e processamento sensorial. A seção de decomposição de tarefas inclui o monitoramento e o processamento da tarefa selecionada para execução. A seção de modelagem do mundo inclui modelos de objetos e estruturas elaborados em sistemas CAD. O processamento sensorial inclui o processamento de sinais, o reconhecimento de padrões, o reconhecimento de objetos e a observação de diferenças entre o mundo e o modelo interno. Na Figura 4.4 é ilustrada a organização desta arquitetura.

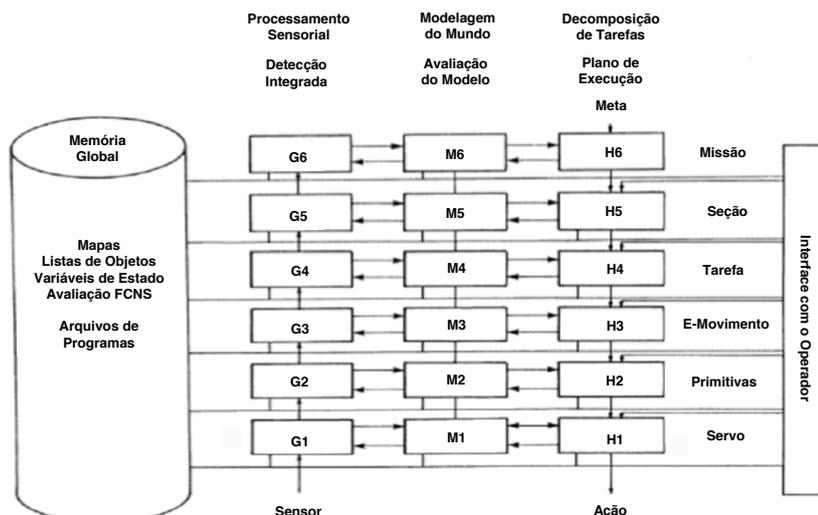


Figura 4.4: Arquitetura NASREM. As setas representam o fluxo de informações dentro da arquitetura. Esta arquitetura possui uma organização hierárquica em diversos níveis e também uma organização seqüencial que possui três seções: decomposição de tarefas, modelagem do mundo e processamento sensorial (Albus et al., 1989).

A arquitetura de BLACKBOARD (Carver and Lesser, 1994) emprega a idéia de módulos distribuídos que se comunicam por meio de uma memória de trabalho. Estes módulos podem ser compostos por sensores, mecanismos de raciocínio e atuadores. O modelo básico desta arquitetura é composto por um *quadro negro*, um grupo de fontes de conhecimento e um mecanismo de controle. O *quadro negro* é um banco de dados compartilhado por todas as fontes de conhecimento e é estruturado de modo hierárquico. Ele contém os dados e as hipóteses formuladas pela arquitetura. As fontes de conhecimento podem ser, entre outras, sistemas de visão computacional, sistemas sensores, sistemas de desvio de obstáculo e sistemas de planejamento. As fontes de conhecimento devem ser independentes de algum modo e somente devem se comunicar por meio do *quadro negro*. O sistema de controle é responsável pela detecção de eventos e pela administração da execução das tarefas. Na Figura 4.5 é ilustrado um exemplo da organização deste tipo de arquitetura. As modificações efetuadas no *quadro negro* são descritas para o monitor deste como um conjunto de eventos. O monitor identifica quais fontes de conhecimento devem ser empregadas por novos eventos e invoca os componentes de pré-condições para o disparo destas fontes. Os componentes de pré-condição que obtiverem sucesso retornam um conjunto de estímulos e respostas para o monitor, que as utiliza para criar uma KSI (representando uma fonte de conhecimento ativa), que é inserida na agenda de execuções. O monitor também atualiza a base de dados de controle baseado nos novos eventos. Desta forma, o gerenciador classifica os KSIs existentes na agenda, remove o de maior prioridade e invoca as fontes de conhecimento e componentes apropriados para a sua execução.

4.3 Arquiteturas Reativas

Em contraste com as arquiteturas deliberativas, as reativas não empregam um modelo do mundo, e minimizam a informação sobre seus estados internos e sobre o ambiente. Os módulos de uma arquitetura reativa podem ser vistos como comportamentos que devem ser apresentados pelo robô em uma interação com o ambiente. O controle reativo é essencialmente um mecanismo reflexivo no qual pares de estímulos-respostas governam as ações. Assim, estas arquiteturas reagem a condições do ambiente, emitindo respostas selecionadas em certos contextos (Arkin, 1989). A principal vantagem desta estratégia de controle é a resposta rápida do robô as mudanças em ambientes dinâmicos, nos quais nenhuma informação a priori está disponível. Arquiteturas reativas requerem pouca memória e não computam ou armazenam representações do mundo. Arquiteturas puramente reativas são muito criticadas

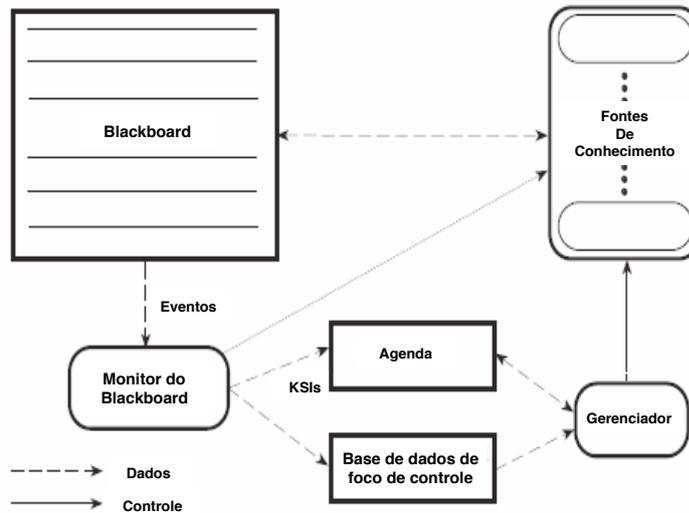


Figura 4.5: Arquitetura BLACKBOARD. As setas representam o fluxo de informações dentro da arquitetura. O modelo básico desta arquitetura é composto por um *quadro negro*, um grupo de fontes de conhecimento e um mecanismo de controle (Carver and Lesser, 1994).

pois apresentam somente mecanismos simples que produzem comportamentos. Entretanto, Brooks (Brooks, 1986) afirma que não existe a necessidade de representação simbólica ou mecanismos complexos para produzir comportamentos inteligentes. Nesta Seção são apresentadas duas das principais arquiteturas reativas encontradas na literatura.

A arquitetura SUBSUMPTION (Brooks, 1986) é baseada em comportamentos puramente reativos e em um método para se reduzir a arquitetura de controle de um robô a um conjunto de comportamentos para a realização de tarefas. Os comportamentos são representados por camadas de controle. Camadas distintas trabalham simultaneamente e assincronamente em metas individuais. Cada comportamento é representado na forma de uma máquina de estado. Cada máquina executa as suas ações e é responsável pelas próprias percepções do ambiente. Não existe uma memória global, barramento ou processo síncrono entre comportamentos e não existe uma representação ou modelo do mundo. Todas as camadas têm acesso direto e constante às informações dos sensores. As camadas são hierarquicamente organizadas, permitindo que camadas superiores inibam ou suprimam os sinais de controle das camadas inferiores. A supressão elimina o sinal de controle de uma camada inferior e o substitui por um sinal de um procedimento de uma camada superior. Quando a saída da camada superior não está ativa, o nó de supressão não afeta o sinal de controle da camada inferior, que então assume o controle do robô. A inibição desliga um sinal de uma camada inferior, não produzindo nenhuma saída para o canal. Por estes mecanismos, as camadas

hierarquicamente superiores podem suprimir as inferiores. Esta hierarquia de camadas constitui o método de coordenação competitiva da arquitetura. Então, os comportamentos podem ser executados simultaneamente, coordenados por meio de um processo de Subsunção, no qual um comportamento inibe outro temporariamente, resultando em diversas combinações de comportamentos. Na Figura 4.6 é ilustrado um exemplo da organização desta arquitetura.

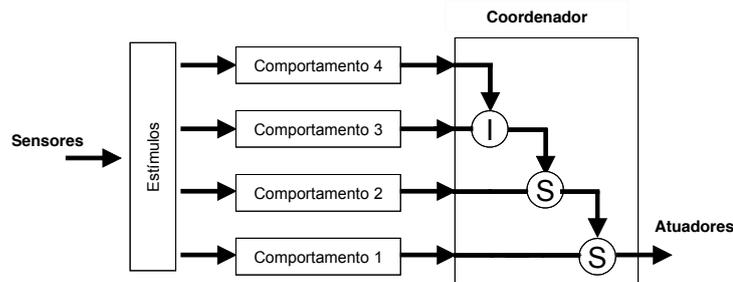


Figura 4.6: Arquitetura SUBSUMPTION. As setas representam o fluxo de informações dentro da arquitetura. Esta arquitetura possui uma organização baseada em camadas hierarquicamente organizadas. Todas as camadas trabalham simultaneamente e assincronamente em metas individuais e as camadas superiores podem suprimir (S) ou inibir (I) uma camada inferior (Arkin, 1998).

A arquitetura MOTOR-SCHEMA (Arkin, 1989) possibilita a decomposição dos comportamentos em esquemas de percepção e de motor. Cada esquema de motor possui um esquema de percepção que provê informação do ambiente. Estes esquemas de percepção podem ser recursivamente definidos, ou seja, um sub-esquema de percepção pode extrair informações usadas por esquemas globais de nível superior e essas informações podem ser processadas subsequente por outros sub-esquemas. A informação produzida por uma sub-esquema de percepção é mesclada por esquemas superiores antes de serem enviadas ao respectivo sub-esquema ou esquema motor. Um esquema motor pode empregar diversas técnicas de Inteligência Artificial para executar ações de acordo com as informações recebidas dos respectivos esquemas de percepção. A saída de um esquema motor de navegação, por exemplo, poderia ser um vetor contendo a orientação e velocidade desejadas para o movimento. O comportamento de navegação poderia ser obtido pela combinação de diversos esquemas, como evitar obstáculos em movimento, evitar obstáculos estáticos, manter-se no caminho e mover-se para o objetivo. Desta forma, o comando de movimento seria o resultado da superposição de todos os esquemas existentes, ou seja, uma soma de vetores. Na Figura 4.7 é ilustrado um exemplo da organização desta arquitetura.

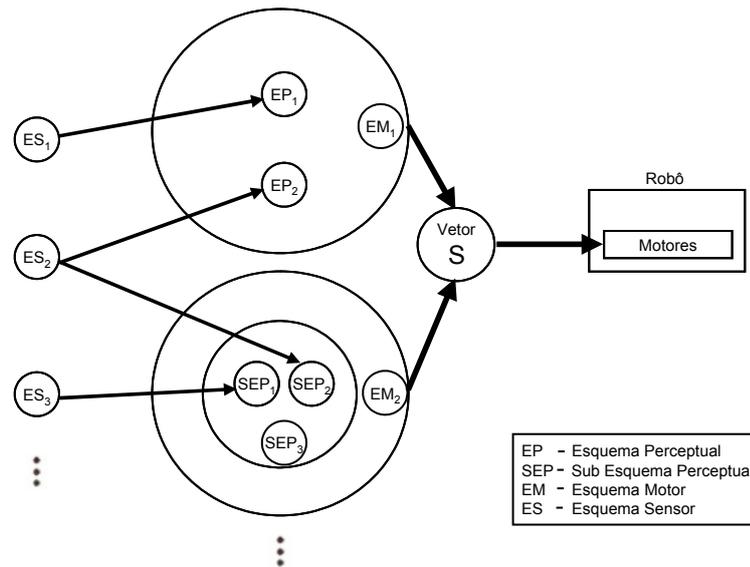


Figura 4.7: Arquitetura MOTOR SCHEMA. As setas representam o fluxo de informações dentro da arquitetura. Esta arquitetura possui uma organização baseada em esquemas de motor e esquemas de percepção que podem ser decompostos em sub-esquemas. Os resultados de dois ou mais esquemas de motor podem ser combinados para produzir um vetor de comando resultante (S) que é então executado pelo robô (Arkin, 1998).

4.4 Arquiteturas Híbridas

As arquiteturas híbridas empregam o melhor das arquiteturas deliberativas e reativas. Elas definem um agente com dois componentes: um deliberativo, empregando um modelo simbólico do mundo e um reativo, capaz de reagir ao ambiente sem qualquer raciocínio simbólico. Normalmente, o componente reativo tem prioridade sobre deliberativo. Deste modo, o robô pode emitir uma resposta rápida na presença de eventos inesperados no ambiente, enquanto executa uma tarefa para alcançar um objetivo de longo prazo. A estrutura destas arquiteturas normalmente é implementada em níveis hierárquicos. Quanto mais alto o nível, mais abstrata é a informação manipulada. Níveis mais altos trabalham com objetivos de mais longo prazo e níveis mais baixos com objetivos de curto prazo (Arkin, 1998) (Oza, 1999). Nesta Seção são apresentadas as principais arquiteturas híbridas propostas na literatura.

A arquitetura ATLANTIS (*Three-Layer Architecture goes Navigating Through Intricate Situations*) (Gat, 1992) é composta por uma camada reativa e uma camada deliberativa. O controlador é um mecanismo reativo responsável pelas atividades primitivas, ou seja, atividades que não necessitam de decisões ou computações. Esta camada de controle recebe os dados dos sensores e envia comandos reativos aos atuadores. O mapeamento estímulo-resposta é efetuado pelo sequenciador. O sequenciador é um sistema operacional de propósito

especial que controla o início e término das atividades primitivas e das atividades deliberativas que são executados no deliberador. Esta camada emprega uma visão de alto nível dos objetivos do robô. Ela é responsável por controlar as seqüências de atividades primitivas e computações deliberativas. Adicionalmente, esta camada deve lidar com as falhas inesperadas nas execuções das atividades. Isto requer a manutenção cuidadosa de grande quantidade de informação do estado interno, pois o sequenciador deve poder lembrar quais ações foram tomadas no passado para decidir quais ações devem ser tomadas no presente. As respostas da camada deliberativa atendem aos pedidos do sequenciador, quando este solicita a execução de tarefas deliberativas. Esta camada é responsável por executar tarefas mais complexas e sofisticadas como planejamento e modelagem do mundo. O deliberador executa tarefas sob o controle do sequenciador. Todas as computações deliberativas são iniciadas (e podem ser terminadas antes de conclusão) pelo sequenciador. Isto permite ao mesmo direcionar os recursos computacionais a uma tarefa necessária. Os resultados das tarefas deliberativas são armazenados em um banco de dados que pode ser acessado pelo sequenciador.

A arquitetura AuRA (*Autonomous Robot Architecture*) (Arkin and Balch, 1997) possui dois componentes: um sistema hierárquico deliberativo composto por um planejador de missão, um raciocinador espacial e um sequenciador de plano; um sistema reativo composto por um controlador baseado em esquemas e diversos esquemas de percepção e de motor. O nível mais alto de AuRA é um planejador de missões responsável por estabelecer metas de longo prazo para o robô, assim como as restrições das ações das mesmas. O raciocinador espacial emprega conhecimento cartográfico armazenado em uma memória permanente para construir uma sucessão de movimentos que o robô deve executar para completar sua missão. O sequenciador de planos traduz cada movimento gerado pelo raciocinador espacial em um conjunto de comportamentos motores para execução da tarefa. Finalmente, o conjunto de comportamentos (esquemas), especificado e instanciado pelo sequenciador de planos, é enviado ao robô para a sua execução. Na Figura 4.8, é ilustrado um exemplo da organização desta arquitetura.

A arquitetura DAMN (*Distributed Architecture is Mobile Navigation*) (Rosenblatt, 1995) é implementada como um grupo de comportamentos distribuídos que se comunicam com um árbitro centralizador de comandos, enviando votos a favor de ações que satisfazem seus objetivos e contra as ações que não os satisfazem. Os esquemas de votos são implementados baseados em restrições, atuadores e efeitos. O árbitro é então responsável por combinar os votos para os comportamentos e gerar as ações que reflitam os seus objetivos e prioridades. Desta forma, esta arquitetura pode integrar módulos de decisão

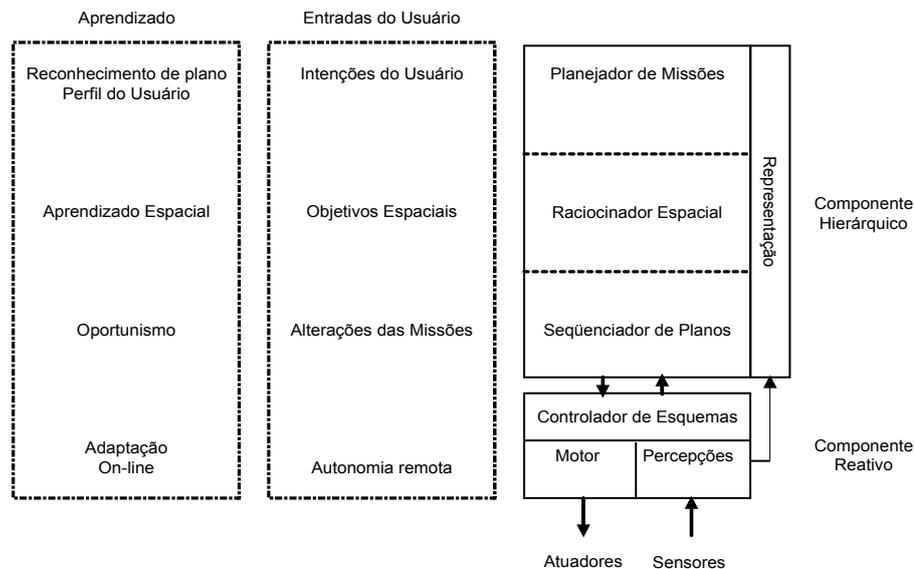


Figura 4.8: Arquitetura AuRA. Esta arquitetura possui dois componentes: um sistema hierárquico deliberativo que consiste em planejador de missão, um raciocinador espacial e sequenciador de plano; um sistema reativo composto por um controlador baseado em esquemas e diversos esquemas de percepção e de motor (Arkin and Balch, 1997).

independentes que se comunicam com o árbitro, suportando a escalabilidade pelo desenvolvimento de novos módulos de comportamento. Na Figura 4.9, é ilustrado um exemplo da organização da arquitetura, no qual os comportamentos individuais como desvio de obstáculos ou manutenção de rota enviam votos ao módulo de arbitragem de comando. Estes votos são então combinados e o comando resultante é enviado ao controlador do robô. Cada comportamento é associado a um peso que reflete sua prioridade relativa em controlar o robô. Um gerenciador também pode ser usado para variar estes pesos durante o curso de uma missão baseado em conhecimento de quais comportamentos seriam pertinentes e seguros em determinadas situações.

Em (Ishiguro et al., 1999), é proposta uma arquitetura robótica composta por módulos situados e reativos, além de diversos componentes para a manutenção destes módulos. Os módulos reativos possibilitam uma rápida resposta em situações de perigo, como uma colisão com um obstáculo. Os módulos situados implementam comportamentos sofisticado ou de alto-nível que são executados de acordo com as tarefas e o ambiente. Estes módulos são executados pelo módulo de controle. O módulo de controle define a ordem de execução dos módulos situados, seleciona módulos executáveis e avalia a execução dos mesmos. Este módulo executa seqüencialmente os módulos situados referenciados em um plano de tarefas. Quando o módulo de controle perde um módulo executável devido a um evento inesperado ou mudança no ambiente, este busca e seleciona um novo módulo situado na rede de módulos

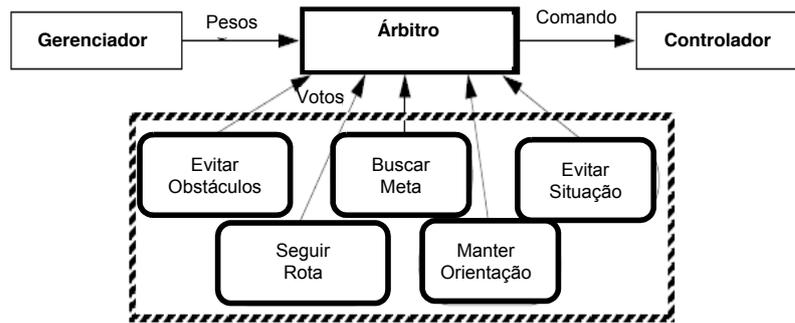


Figura 4.9: Arquitetura DAMN. Esta arquitetura é organizada como um grupo de comportamentos distribuídos que se comunicam com um árbitro centralizador de comandos, enviando votos a favor de ações que satisfazem seus objetivos e contra as ações que não os satisfazem. Estes votos são então combinados e o comando resultante é enviado ao controlador do robô (Rosenblatt, 1995).

situados. Para a seleção de um módulo, o avaliador considera os resultados previstos com a execução das tarefas do mesmo. As representações internas entre os módulos situados são utilizadas não somente para o planejamento, mas para encontrar e recuperar módulos situados adjacentes e apropriados a determinadas situações. Adicionalmente, se algum sensor detecta alguma situação caracterizada como perigosa, enquanto os módulos situados estão sendo executados, o módulo reativo assume o controle do robô e este age reativamente. Na Figura 4.10, é ilustrado um exemplo da organização desta arquitetura.

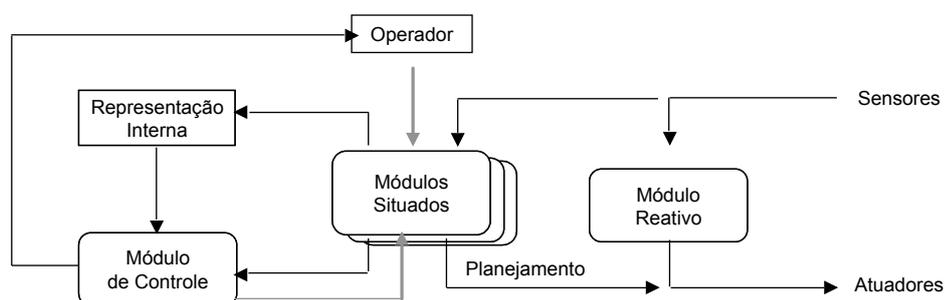


Figura 4.10: Arquitetura baseada em Módulos Situados. As setas representam o fluxo de informações dentro da arquitetura. Esta arquitetura é composta por módulos situados e diversos componentes para a manutenção dos módulos situados como o executor, o módulo reativo, o avaliador e o planejador (Ishiguro et al., 1999).

4.5 Arquiteturas de Controle de Robôs Sociáveis

As arquiteturas de controle de robôs sociáveis possuem mecanismos e estruturas especiais para tratar as complexidades da personificação social e as funcionalidades sociais exigidas para estabelecer as interações com seres humanos de forma natural. Estas arquiteturas devem tratar os problemas de integração de componentes deliberativos e reativos com a integração de componentes sociais que controlem a personificação, a interação e o aprendizado social. Nesta Seção, são apresentadas algumas das principais arquiteturas de controle de robôs sociáveis existentes, que serviram como base para este trabalho, além de algumas arquiteturas propostas recentemente na literatura.

A arquitetura EGO (*Emotionally GrOuded Architecture*) (Arkin et al., 2001) (Arkin et al., 2003) é uma arquitetura para o desenvolvimento de sistemas comportamentais baseados em teorias da psicologia e etologia. Um modelo etológico específico foi criado com base nos estudos dos comportamentos dos cachorros domésticos. A arquitetura utiliza uma abordagem *botton-up* e *top-down* para o desenvolvimento do sistema de comportamento. Os comportamentos são organizados em níveis hierárquicos de abstração, no qual os comportamentos mais abstratos estão em níveis hierárquicos mais altos. A arquitetura possui um componente motivacional que recebe os estímulos detectados no ambiente. Desta forma, o sistema utiliza um processo de motivação para organizar e manter a seqüência dos comportamentos. Este componente utiliza um modelo emocional e um modelo de instinto. O modelo emocional incorpora seis emoções básicas propostas em (Ekman, 1999): felicidade, raiva, tristeza, medo, surpresa e aversão. Adicionalmente, outras seis variáveis agem para manter o estado interno do robô dentro dos limites de valores normais. Estas são denominadas variáveis de instinto que incluem fome, sede, eliminação, fadiga, curiosidade e afeto. O sistema também utiliza um grupo de estruturas perceptuais-motoras para relacionar estímulos e respostas. Ainda, ele possui um componente de ativação em cada camada da hierarquia de comportamentos. Nesta arquitetura, o componente de ativação computa sua saída utilizando os estímulos percebidos no ambiente. O componente motivacional computa sua saída utilizando o modelo emocional e o modelo de instinto. Então, um módulo de seleção de ações seleciona um comportamento, utilizando a saída do componente de ativação e do componente motivacional. A computação é executada a partir dos comportamentos em um nível de organização mais alto. Um mecanismo de inibição lateral é utilizado para evitar conflitos entre comportamentos, de forma que o sistema selecione somente um único comportamento para ativação. A seleção dos comportamentos é executada da camada mais alta para a mais baixa, na qual um comportamento apropriado

é selecionado e então enviado para uma máquina de estados que possui uma seqüência específica de comandos e transições para exibir o comportamento selecionado. Na Figura 4.11 é ilustrada a organização desta arquitetura.

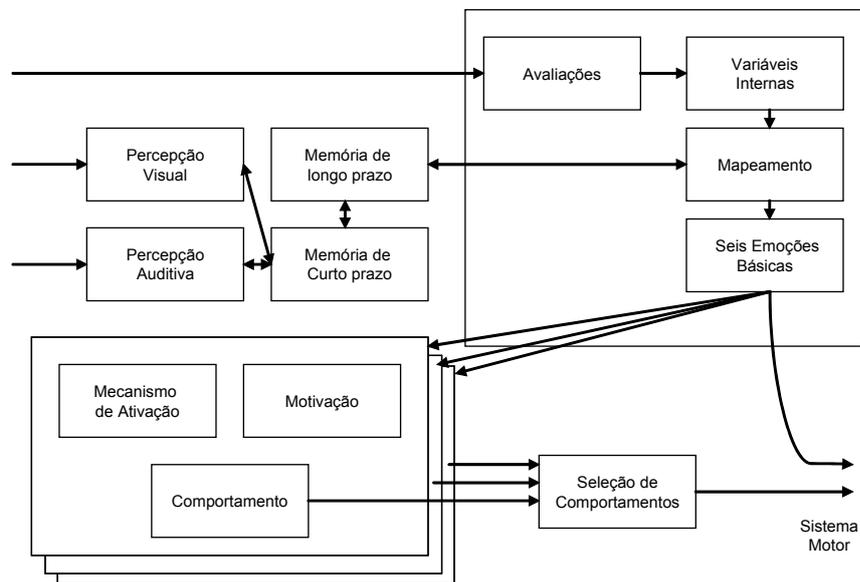


Figura 4.11: Arquitetura EGO (*Emotionally GrOuded Architecture*). Esta arquitetura é composta por um módulo motivacional e comportamentos organizados em camadas de abstrações. As setas indicam o fluxo de informações dentro da arquitetura. O módulo de seleção de comportamentos utiliza a saída do módulo motivacional e do módulo de ativação de cada camada de abstração para selecionar um comportamento apropriado (Arkin et al., 2003).

O robô Kismet, apresentado em (Breazeal, 2002) (Breazeal, 2003b) possui uma arquitetura formada por um sistema de percepção, um sistema emocional, um sistema de necessidades ou direcionadores, um sistema de comportamento e um sistema motor. Esta arquitetura torna possível ao robô exibir expressões faciais que permitem aos seres humanos determinar seu estado interno. O sistema de percepção integra percepções visuais como movimento; cores e faces humanas, junto com efeitos de habituação. Cada percepção visual gera um mapa de características que são combinadas para criar um mapa de ativação que é utilizado para direcionar a atenção do robô. Este sistema influencia e é influenciado pelos sistemas de comportamento e o sistema emocional, provendo um mecanismo de atenção dependente do contexto do ambiente e do estado interno do robô. O sistema emocional é composto de dois sub-sistemas: um sub-sistema que implementa um conjunto de necessidades e um sub-sistema que implementa um conjunto de emoções e estados de expressões. O sub-sistema de necessidades influencia a seleção do comportamento do robô e as expressões do mesmo. O sub-sistema de emoções e estados de expressão influencia as expressões do robô e tem a função de

regular as interações com seres humanos. Eventos externos, como estímulos visuais e sonoros, são detectados pelo robô e são filtrados por diversos extra-atores de característica (cor, movimento, face, voz). No sistema de percepção, estas características são utilizadas por processos de ativadores que codificam o estado interno e externo do robô. O resultado é um conjunto de ativadores de resposta específica que servem como condições antecedente para respostas emotivas específicas. Os ativadores são passados a uma fase de avaliação, na qual eles são rotulados com uma informação, composta por níveis de estimulação, valência e posição, denotado por [A; V; S] na Figura 4.12. Todas as entradas ativas na fase de avaliação são filtradas pelos elicitadores emocionais, para cada emoção existente. Na fase de arbitragem de emoções, os processos de emoção competem para ativação dentro um esquema competitivo (*winner-takes-all*). O vencedor evoca uma expressão facial correspondente, postura de corpo e verbalização, enviando os valores de [A;V; S] ao sistema motor. O vencedor pode também ativar uma resposta de comportamento correspondente enviando a energia de ativação para o sistema de comportamento. Na Figura 4.12 é ilustrada a organização desta arquitetura.

A arquitetura SRA (*Social Robot Architecture*) (Duffy et al., 2005) é uma arquitetura híbrida para robôs sociáveis composta por uma camada reativa, uma camada deliberativa, um mecanismo de raciocínio social e uma camada que efetua a interface entre a arquitetura e o robô físico. A SRA foi projetada com funcionalidade social explícita para interações entre robôs e interações entre robôs e seres humanos. Os componentes fundamentais desta arquitetura são a camada de abstração de hardware para aplicações em plataforma robóticas heterogêneas, a integração entre as camadas reativas e deliberativas, e um mecanismo de intenções-convicções-desejos para apoiar interações sociais explícitas. O nível físico provê portabilidade da arquitetura em relação a estrutura física do robô. A camada reativa supervisiona a camada física, administrando uma biblioteca de módulos primitivos de atividades e comportamentos. As atividades são normalmente responsáveis pela aquisição de dados e processos sensoriais. O módulos de comportamento implementam respostas reflexas a eventos inesperados ou perigosos, além de ações mais complexas. Estas constituem um conjunto de habilidades primárias de sobrevivência do robô. Normalmente estes módulos tentam manter ou estabelecer relações simples entre estímulos sensoriais e o estado interno do robô, ou executar sucessivas operações simples. O nível deliberativo da SRA segue um esquema de organização multi-agente, com diversos agentes que supervisionam os diferentes níveis funcionais do robô. Em um determinado momento, diversos agentes compartilham o controle da plataforma robótica. Estes agentes variam em complexidade, desde módulos com conhecimentos processuais

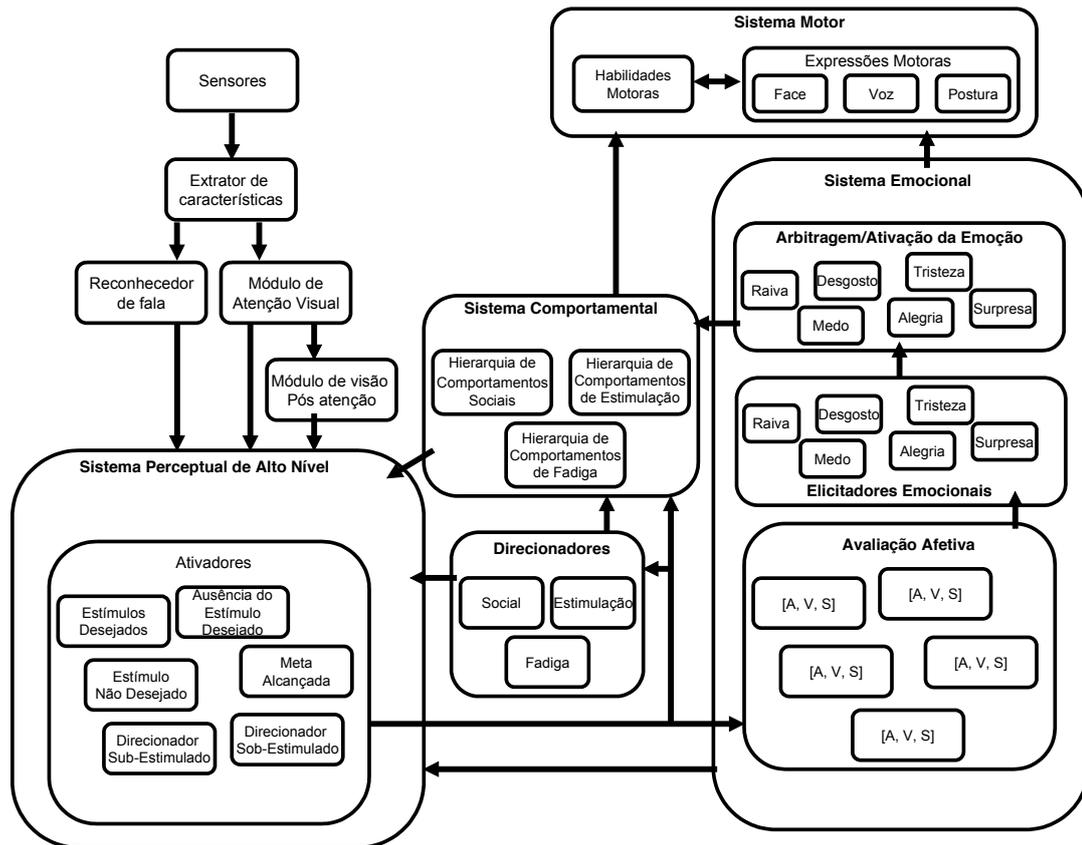


Figura 4.12: Arquitetura do robô Kismet. As setas indicam o fluxo de informações dentro da arquitetura. Esta arquitetura é composta por um sistema de percepção, um sistema emocional, um sistema de necessidades ou direcionadores, um sistema de comportamento e um sistema motor. As setas indicam o fluxo de informações dentro da arquitetura. O sistema de percepção do robô extrai as características salientes do ambiente, baseado nas necessidades internas providas pelo sistema motivacional. Essas características são então usadas pelo sistema emocional e pelo sistema de necessidades para atualizar o estado interno do robô e influenciar na seleção dos comportamentos que serão exibidos pelo mesmo (Breazeal, 2003b).

simples que lidam com capacidades de baixo nível da plataforma (como sensores), até módulos com capacidades mais complexas de raciocínio. O nível social provê mecanismos para o controle das interações com outros robôs e com seres humanos, por meio de uma linguagem de comunicação entre agentes (LCA) e por mecanismos especificados na arquitetura FIPA (www.fipa.org). A LCA provê o tratamento de mensagens que podem disparar regras de comportamento que são enviadas ao nível deliberativo para serem integradas à seqüência atual de tarefas em execução. Estas tarefas são então passadas ao nível reativo que arbitra a execução das mesmas enquanto nenhum perigo ou evento inesperado for detectado. Na Figura 4.13 é ilustrada a organização desta arquitetura.

A arquitetura AD (*Automatic-Deliberative*) (Salichs et al., 2006) é uma ar-

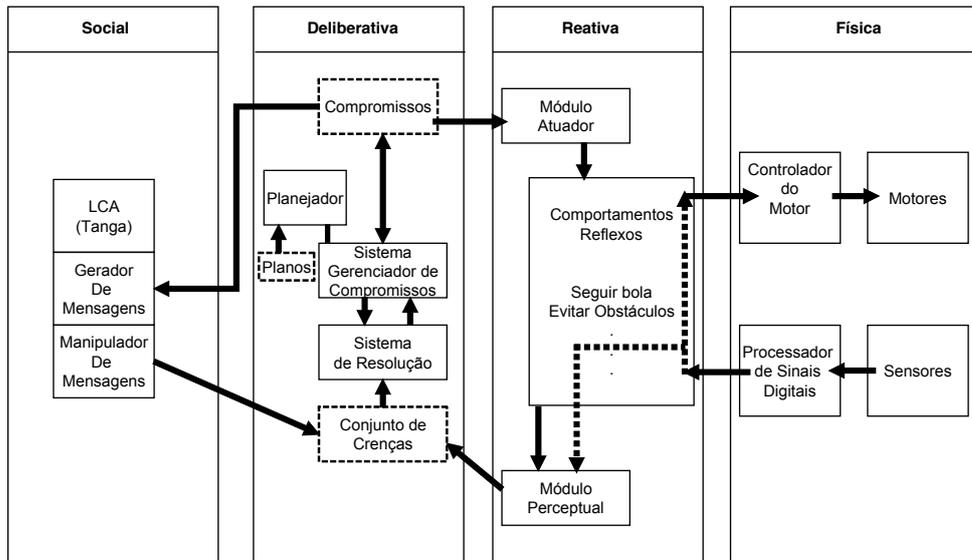


Figura 4.13: Arquitetura SRA (*Social Robot Architecture*). Esta arquitetura é composta por uma camada reativa, uma camada deliberativa, um mecanismo de raciocínio social e uma camada que efetua a interface entre a arquitetura e o robô físico. As setas indicam o fluxo de informações dentro da arquitetura. Durante sua operação, a LCA prove o tratamento de mensagens que podem disparar regras de comportamento que são, então, enviadas ao nível deliberativo para serem integradas à sequência atual de tarefas sendo executadas. Estas tarefas são então passadas ao nível reativo que arbitra a execução das mesmas enquanto nenhum perigo ou evento inesperado for detectado (Duffy et al., 2005).

Arquitetura híbrida baseada em habilidades. Uma habilidade representa a capacidade do robô para executar uma tarefa particular. Elas representam capacidades de ações e percepções pré-desenvolvidas. Esta arquitetura é composta por duas camadas, uma deliberativa e uma reativa (automática). No nível deliberativo, encontram-se habilidades capazes de executar tarefas de alto nível, enquanto no nível reativo existem habilidades responsáveis por interagir com o ambiente. O planejador de caminho, a modelagem do mundo e o supervisor de tarefa são algumas das habilidades incluídas no nível deliberativo. No nível reativo, encontram-se as habilidades sensório-motoras e as habilidades perceptivas. As primeiras são responsáveis pelo movimento do robô. As segundas são responsáveis por detectar eventos necessários para produzir as transições do sequenciador que gerencia as tarefas executadas pelo robô. Adicionalmente, um Sistema de Controle Emocional (SCE) influencia as deliberações da arquitetura. Dentro do SCE, existem três módulos diferentes: Direcionadores, Módulo de Seleção de Atividade e Sistema Supervisor Emocional (SSE). O módulo de Direcionadores é o que controla as necessidades básicas do robô. O módulo de Seleção de Atividade, por outro lado, determina metas e tendências de ações para o robô. O módulo SSE gera o estado emocional do robô. O

papel do SSE na arquitetura de controle é estabelecer metas gerais como *bem estar*, que influenciam na seleção de comportamentos e na avaliação global da situação atual. Esta avaliação global também pode ser usada como reforço para os processos de aprendizado. Quando uma mudança altera o valor de alguma variável do SSE para um valor maior que um certo limiar, uma emoção torna-se ativa. Estas emoções incluem felicidade, raiva, medo e tristeza. Neste sentido, o SSE age como um sistema de alarme no qual as emoções são os sinais de alerta. Desta forma, quando o nível deliberativo detecta um evento enviado pelo nível automático, seu sequenciador de tarefas seleciona novas habilidades, considerando a influencia do SCE. Então, uma ordem de execução desta habilidade é passada ao nível reativo que executa a tarefa solicitada enquanto monitora o ambiente para detectar eventos inesperados. Na Figura 4.14 é ilustrada a organização desta arquitetura.

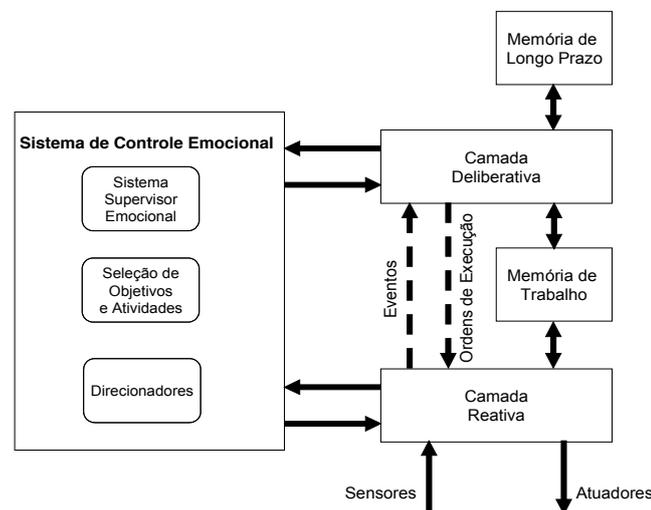


Figura 4.14: Arquitetura AD (*Automatic-Deliberative*). Esta arquitetura é baseada em habilidades e é composta por uma camada reativa, uma camada deliberativa e e por um sistema de controle emocional. As setas indicam o fluxo de informações dentro da arquitetura. Durante sua operação, quando o nível deliberativo detecta um evento enviado pelo nível automático, seu sequenciador de tarefas seleciona novas habilidades, considerando a influencia do SCE, e então uma ordem de execução desta habilidade é passada ao nível reativo que executa a tarefa solicitada, enquanto monitora o ambiente para detectar eventos inesperados (Salichs et al., 2006).

Em (Breazeal et al., 2005) (Breazeal et al., 2006) é apresentada uma arquitetura para o aprendizado e imitação de expressões faciais humanas, que possui uma arquitetura composta por um sistema sensor, um sistema de percepção, um sistema de ações e um sistema motor. O sistema de percepção utiliza um mecanismo hierárquico chamado árvore de percepção para extrair informação do ambiente a partir dos sensores. Cada nó da árvore é chamado de percepção, com as percepções mais específicas mais próximas aos

nós folhas. Percepções são unidades de percepção atômicas cujo trabalho é reconhecer e extrair características a partir dos sensores. Esta arquitetura também possui um sistema de percepção que recebe entradas sensoras do software de *Axiom fffT* e implementa diversas percepções simples. Existe uma unidade de percepção de face que torna-se ativa sempre que o *Axiom fffT* indica a presença de uma face humana. Este sistema possui unidades de percepção que corresponde a órgãos faciais como sobrancelhas, olhos, nariz e boca. Existem diversas unidades de percepção de movimento que detectam quando as características faciais do ser humano mudam, além de unidades de contingência que detectam quando elas mudam com respeito aos próprios movimentos do robô. O sistema de ação do robô é responsável pela arbitragem de comportamento, selecionando qual comportamento exibirá e quando este será exibido. Os comportamentos são representados individualmente como registros de ação. Os registros de ação são compostos por uma ação e seu contexto de ativação. A ação é um segmento de código responsável por enviar solicitações de movimentos ou seqüências de movimentos para o sistema motor do robô. O sistema motor é responsável por executar os movimentos exigidos para exibir um comportamento selecionado pelo sistema de ações. Nesta arquitetura, os movimentos motores são representados como caminhos por um grafo dirigido com pesos, conhecido como o grafo de poses da criatura. Cada nó (ou pose) no grafo é associado a uma configuração das juntas do robô e pode ser visto como uma única configuração do corpo do mesmo. Uma conexão entre duas poses representa uma transição permitida entre as configurações motoras destas. Estas ligações são projetadas para permitir somente movimentos biologicamente plausíveis e seguros que não coloquem os mecanismo do robô em risco ou apresentem poses que não sejam naturais. As poses e as conexões entre elas definem o espaço de possíveis movimentos do robô, com trajetórias completas de movimento existindo como rotas por este espaço. Desta forma, um comportamento exibido pelo robô pode representar um caminho, dentro do grafo, que passa por diversos nós ou poses. Além do grafo de poses, o sistema motor contém diversas rotinas que são capazes de gerar conexões dentro do espaço de poses, com respeito aos pedidos de ações. Nesta arquitetura, o robô aprende a mapear as expressões faciais percebidas a seu espaço intermodal (seu próprio espaço de juntas), por meio da imitação de seus movimentos por um ser humano. O robô gera uma variedade de posições motoras pela exploração de seus movimentos. Quando o ser humano exibe expressões em resposta ao movimento do robô, este decide se está sendo imitado e utiliza a expressão atual do humano e sua própria expressão atual para treinar um conjunto de redes neurais que são então utilizadas para mapear a expressão do ser humano ao espaço intermodal do robô. Uma

vez que estas redes estejam treinadas para codificar este mapeamento, o robô pode converter estes dados em sua representação intermodal e pode classificar uma pose, apresentada por um ser humano, como uma de suas próprias. Isto permite ao robô produzir uma pose semelhante, imitando as expressões do ser humano. Na Figura 4.15 é ilustrada a organização desta arquitetura.

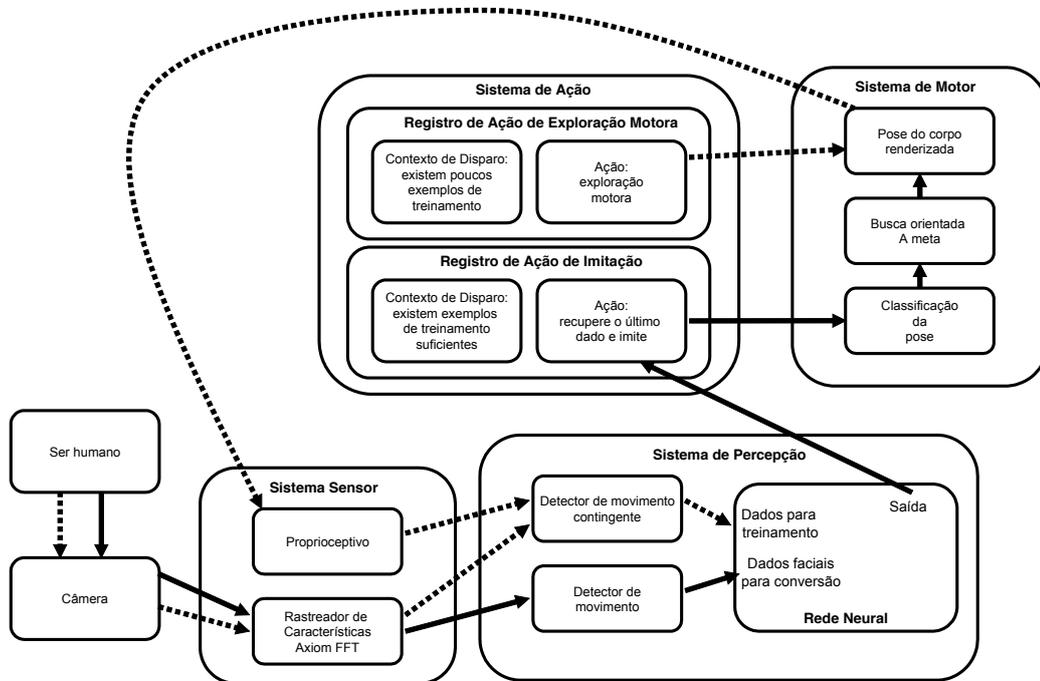


Figura 4.15: Arquitetura do Robô Leonardo. composta por um sistema sensor, um sistema de percepção, um sistema de ações e um sistema motor. As setas sólidas indicam o fluxo de informações durante uma imitação de uma expressão. As setas pontilhadas indicam o fluxo de informações durante uma exploração motora. O robô utiliza a expressão atual do humano e sua própria expressão atual para treinar um conjunto de redes neurais utilizadas para mapear a expressão do ser humano em seu espaço intermodal, permitindo que o robô aprenda a produzir uma pose semelhante, imitando as expressões do ser humano (Breazeal et al., 2005).

4.6 Considerações Finais

A construção de robôs sociáveis é um grande desafio. Estes robôs podem ser aplicados em diversos domínios e ambientes e seus projetos podem variar muito em termos de técnicas, funcionalidades e objetivos sociais. Como apontado anteriormente, o projeto e construção de robôs sociáveis pode se beneficiar muito do emprego de uma arquitetura robótica com estruturas e mecanismos apropriados para esta classe de aplicação.

As arquiteturas de controle de robôs sociáveis devem ser capazes de tratar os problemas de integração de componentes deliberativos e reativos, com com-

ponentes sociais que controlem a personificação, a interação e o aprendizado social. Um exame das principais arquiteturas de controle de robôs sociáveis apresentadas mostra que um elemento chave destas arquiteturas é um sistema capaz de simular emoções ou estados internos do robô. Este sistema confere pró-atividade ao robô, permitindo ao mesmo interagir com o ambiente de forma a satisfazer suas necessidades e controlar possíveis alterações em seu estado interno ao invés de apenas reagir aos eventos como um organismo passivo e reativo. Pode-se citar também como elemento chave um sistema de percepção capaz de detectar faces humanas, elementos salientes que apresentam padrões complexos e objetos em movimentos. Estas funcionalidades permitem a estes sistemas simular as preferências visuais das crianças com até 18 meses de idade, conferindo à arquitetura a capacidade de detectar e dirigir todo o processo cognitivo e motor em resposta a estímulos socialmente importantes. Outros elementos comuns a estas arquiteturas são os sistemas comportamental e motor, muitas vezes organizados hierarquicamente, capazes de selecionar e exibir comportamentos naturais em diferentes contextos do ambiente. Estas arquiteturas também podem apresentar mecanismos de aprendizado por meio da interação, podendo se diferenciar em aprendizado por tutela, por imitação ou por contingência.

Nesse Capítulo foram apresentadas algumas das principais arquiteturas robóticas propostas na literatura. Uma importante característica que difere as arquiteturas de controle de robôs sociáveis das demais arquiteturas existentes são os mecanismos e estruturas especiais para tratar as complexidades da personificação social e as funcionalidades sociais exigidas para estabelecer as interações com seres humanos de forma natural. Neste contexto, os processos de aprendizado e modelagem do comportamento humano evidenciados na *Análise do Comportamento*, uma importante teoria da Psicologia, pode levar ao desenvolvimento de métodos e estruturas promissores para a construção de robôs sociáveis capazes de aprender pela interação com o ambiente e capazes de exibir comportamentos apropriados.

Uma vez apresentada a revisão bibliográfica realizada durante a pesquisa, no próximo Capítulo é apresentada a proposta deste trabalho, destacando as diferenças entre este e os principais trabalhos relacionados existentes na literatura, assim como as técnicas computacionais empregadas para o desenvolvimento da arquitetura proposta.

Arquitetura Proposta

5.1 *Considerações Iniciais*

Após a pesquisa bibliográfica sobre robôs sociáveis, apresentada no Capítulo 2, formulou-se uma hipótese de pesquisa segundo a qual os processos de aprendizado e modelagem do comportamento humano, evidenciados na Análise do Comportamento, podem levar ao desenvolvimento de métodos e estruturas promissoras para a construção robôs sociáveis capazes de aprender pela interação com o ambiente e capazes de exibir comportamento social apropriado. Desta forma, foi realizada uma a pesquisa sobre a teoria da Análise do Comportamento, apresentada no Capítulo 3, com o objetivo de embasar teoricamente este trabalho, assim como buscar inspiração para o desenvolvimento das estruturas e mecanismos necessários. Adicionalmente, foram explorados os principais trabalhos relacionados existentes na literatura, apresentados no Capítulo 4. Estas pesquisas bibliográficas realizadas levaram à definição do escopo deste trabalho e à definição da arquitetura proposto, além de algumas técnicas computacionais e para a implementação das estruturas e mecanismos necessários para a realização do trabalho proposto.

Neste Capítulo é apresentada a proposta deste trabalho, com a descrição da arquitetura desenvolvida. Adicionalmente, neste Capítulo são apresentadas as técnicas computacionais e algoritmos empregados para a construção da arquitetura proposta..

Este Capítulo está organizado da seguinte maneira. Na Seção 5.2, é apresentada a descrição da arquitetura desenvolvida, à luz dos principais trabalhos relacionados. Na Seção 5.3, são apresentadas as técnicas computacionais empregadas para a implementação das estruturas e mecanismos da arquitetura.

tura proposta. Na Seção 5.4, são brevemente introduzidas algumas técnicas empregas pelos sistemas de apoio da arquitetura proposta, como o sistema de visão e o sistema de reconhecimento de fala. Finalmente, na Seção 5.5, são feitas as considerações finais deste Capítulo.

5.2 Arquitetura Proposta

Como apontado anteriormente nesta Tese, um grande desafio da robótica sociável é o desenvolvimento de robôs que possuam a noção de socialização, que possam desenvolver habilidades sociais e que possam mostrar empatia e entendimento do mundo real. Esses robôs ainda representam um objetivo distante e o alcance deste requer a contribuição de outras áreas do conhecimento, como a Psicologia, a Ciência Cognitiva e a Sociologia (Dautenhahn, 1997) (Dautenhahn and Billard, 1999) (Scassellati, 2000).

Para interagir com os seres humanos, os robôs precisam *perceber* e *entender* a riqueza do comportamento humano. Portanto, estes robôs devem possuir sistemas de percepções visuais e auditivas que permita a interação em tempo real. Neste contexto pode-se destacar a arquitetura do robô Kismet (Breazeal, 2002) (Breazeal, 2003b) e do robô Leonardo (Breazeal et al., 2005) (Breazeal et al., 2006), que possuem sistemas de percepção orientados para o ser humano.

Nas interações humano-robô, outro grande desafio é o desenvolvimento de mecanismos eficientes que permitam ao robô compartilhar com uma pessoa a atenção sobre um objeto ou evento do ambiente, caracterizando uma habilidade denominada *atenção compartilhada*. Neste ponto, deve-se apontar que as principais arquiteturas sociáveis existentes possuem mecanismos de atenção compartilhada pré-programados, mas não são capazes de aprender esta habilidade social.

Adicionalmente, robôs sociáveis precisam agir pró-ativamente durante as interações com o ambiente. O desenvolvimento de sistemas motivacionais artificiais possibilita que o robô não seja apenas um *escravo* do ambiente e da situação, possibilitando que ele responda a certos estímulos de acordo com suas necessidades internas. Neste aspecto, muitas das principais arquiteturas apresentadas no Capítulo 4 apresentam sistemas motivacionais artificiais capazes de simular necessidades juntamente com emoções.

Além da comunicação e da interação, um robô que co-existe diariamente com pessoas deve ser capaz de aprender e se adaptar a novas experiências. Neste contexto, a maioria das arquiteturas sociáveis existentes na literatura não suportam aprendizado ou suportam somente uma técnica ou modo de aprendizado limitado.

Finalmente, deve-se destacar que as arquitetura principais sociáveis existentes utiliza representações neurais do conhecimento, tornando difícil a interpretação do mesmo por parte dos usuários.

Com base no exposto, nesta Seção é descrita a arquitetura proposta neste trabalho, integrando mecanismos de percepção e aprendizagem, além de uma representação alternativa do conhecimento adquirido e mantido pela mesma. Esta arquitetura integra um sistema de visão computacional, um sistema de reconhecimento de fala, um sistema motor desenvolvido para a cabeça robótica interativa utilizada neste trabalho, e os módulos básicos de controle inspirados na Análise do Comportamento. Adicionalmente, esta arquitetura integra um mecanismo capaz de simular o aprendizado de conceitos sobre objetos do mundo real apresentados ao robô, por meio da tutelação de um ser humano. Nesta arquitetura, destacam-se, além da representação do conhecimento, os dois mecanismos de aprendizagem reunidos em uma única arquitetura, constituídos por um mecanismo capaz de simular o condicionamento operante e um mecanismo capaz de simular o aprendizado por tutelação. O aprendizado por tutelação é um processo de colaboração entre um professor e um aprendiz. Neste processo, um professor direciona a atenção do aprendiz e este, por sua vez, contribui com o processo revelando seu estado de conhecimento e sua compreensão sobre os conceitos apresentados, direcionando o professor. Na Figura 5.1 é apresentada a arquitetura desenvolvida.

O sistema de visão é composto por um módulo de reconhecimento de face capaz de estimar a pose da cabeça de um ser humano, baseado em modelos adaptativos de visão computacional baseados em aparência (Morency et al., 2003). O sistema de visão também é composto por um módulo de detecção de objetos, baseado em modelos de saliência e atenção visual (Itti et al., 1998). Este sistema é capaz de simular algumas preferências visuais de crianças entre os 6 e os 18 meses de idade, como cores salientes e faces humanas. A implementação do sistema de visão é baseado em mapas de características processados para cada percepção (cores e faces).

O sistema de voz é composto por um sistema de reconhecimento de fala e por um sistema de vocalização. O sistema de reconhecimento de fala é capaz de reconhecer a linguagem natural falada no idioma Português Brasileiro e é baseado no Sistema Nuance (Nuance, 2001). Este sistema contém uma máquina de reconhecimento de fala e uma base de conhecimento gramatical. Adicionalmente, existe um módulo reconhecedor implementado em linguagem Java, por permitir um fácil interfaceamento com o sistema *NuanceTM*. Este módulo reconhecedor recebe a codificação da fala, relativa à gramática configurada na base de conhecimento e a envia para o sistema de controle da cabeça interativa por meio de uma porta *socket*.

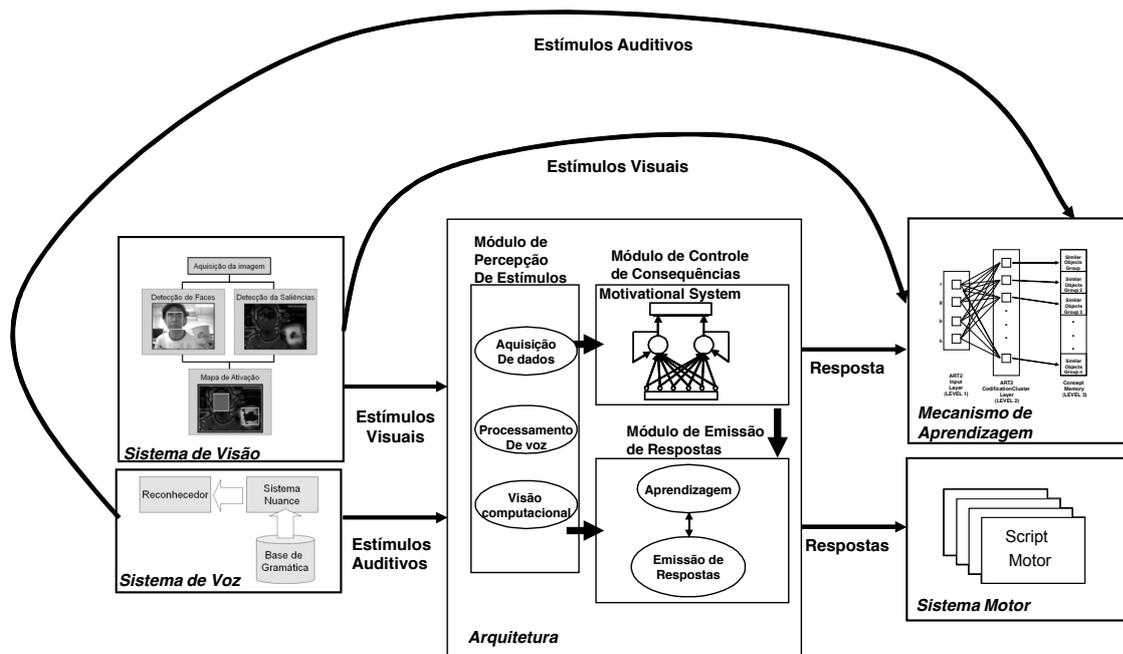


Figura 5.1: Arquitetura geral do sistema de controle do robô. As setas indicam o fluxo de informações entre os módulos do sistema de controle. Este sistema emprega os diversos módulos desenvolvidos durante este projeto de pesquisa: sistema de visão, sistema de voz, mecanismo de aprendizagem por tutoria, sistema motor e arquitetura robótica.

O sistema motor da cabeça robótica interativa foi implementado empregando-se as funções disponibilizadas no SDK do robô. Este sistema motor possui scripts com diversos comandos temporizados de motor, que permitem ao robô emitir os comportamentos necessários aos experimentos realizados.

O mecanismo de aprendizagem proposto neste trabalho de pesquisa é capaz de associar estímulos visuais e auditivos para simular a aprendizagem de conceitos sobre objeto do mundo real por meio da tutoria de um ser humano. Este mecanismo utiliza o sistema de visão para extrair as características visuais de um determinado objeto, composto pelos valores mais frequentes dos canais r, g, e b (do espaço de cores RGB), e do canal h (do espaço de cores HSI). O mecanismo de aprendizagem também utiliza o sistema de voz para adquirir o nome falado do objeto apresentado ao robô. Estas características são então aprendidas e organizadas empregando uma rede neural do tipo ART2 (Carpenter and Grossberg, 1987) e uma memória plana que armazena as características visuais juntamente com o nome falado do objeto para formar novos conceitos.

A arquitetura básica é composta por três módulos principais: percepção de estímulos, emissão de respostas e controle de conseqüências. O módulo de percepção de estímulos emprega os sistemas de voz e visão e é responsável

por detectar o estado do ambiente e codificar este estado empregando uma representação apropriada. O módulo de controle de conseqüências é composto por um *sistema motivacional* que simula as necessidades internas de um indivíduo e detecta os reforços recebido do ambiente. O módulo de emissão de respostas é composto por um mecanismo de aprendizagem capaz de construir uma política não determinística para a emissão de respostas, isto é, quais respostas podem ser emitidas na presença de certos estímulos no ambiente.

Adicionalmente, a representação de conhecimento adotada para a arquitetura básica é baseada em uma representação relacional de primeira ordem (Driessens, 2004) (Otterlo, 2005), possibilitando a representação e a manipulação de grandes espaços de busca e decisão de um modo econômico, em relação a um algoritmo de aprendizado por reforço clássico.

A seguir, são apresentadas as técnicas empregadas para a construção dos módulos componentes da arquitetura proposta.

5.3 Técnicas e Algoritmos

Nesta Seção são introduzidas as técnicas de Inteligência Artificial empregadas para o desenvolvimento da arquitetura propostas neste projeto de pesquisa.

5.3.1 Aprendizado por Reforço

A teoria da Análise do Comportamento demonstra que os organismos vivos são capazes de aprender por meio de interações com o ambiente. O aprendizado por meio de interações é um dos fundamentos de todas as teorias do aprendizado e da inteligência (Sutton and Barto, 1998). Nesse processo de aprendizado, os organismos recebem estímulos reforçadores e punições do ambiente ou de outros organismos em resposta às suas ações. Esses estímulos podem modificar os comportamentos desses organismos, reforçando ou inibindo determinados comportamentos.

Nesse contexto, o Aprendizado por Reforço (AR) surge como um modelo promissor para a simulação desse aprendizado por interações com o ambiente. O AR é um modelo para mapear ações e situações, de maneira a maximizar o valor do estímulo positivo fornecido pelo ambiente, baseado no mesmo princípio do condicionamento operante: o princípio do reforço.

Nesse modelo, não é informado ao agente quais as ações devem ser tomadas em determinadas situações, como em muitos métodos de aprendizado da Inteligência Artificial. O agente precisa descobrir quais ações tomadas em determinados contextos do ambiente resultam em uma maior recompensa. O

AR é definido por não caracterizar métodos de aprendizado, mas por caracterizar um problema de aprendizado. Esse modelo possui os seguintes elementos (Sutton and Barto, 1998):

- Um conjunto discreto de estados do ambiente, S ;
- Um conjunto discreto de ações a serem selecionadas e executadas pelo agente, A ;
- Um conjunto de valores escalares que representam os sinais de reforço do ambiente, \mathfrak{R} ;
- Uma política $\pi : S \rightarrow A$, que mapeia um determinado estado ($s \in S$) do ambiente em uma ação ($a \in A$), definindo o comportamento do agente segundo o contexto do ambiente.
- Uma função de reforço $r : S, A \rightarrow \mathfrak{R}$, que determina a resposta do ambiente para cada ação do agente, mapeando os estados e ações em um valor escalar denominado sinal de reforço.
- Uma função de avaliação $V : S \rightarrow \mathfrak{R}$ ou $Q : S, A \rightarrow \mathfrak{R}$ que estima quais ações são mais adequadas a longo prazo. Essa função prediz a soma dos reforços obtidos ao longo de todos os estados que o agente potencialmente deve visitar a partir do estado atual.

Durante do processo de aprendizado, o agente recebe como entrada uma indicação do estado atual do meio s_t e o sinal de reforço r_t . Então, o agente seleciona uma ação a_t e a executa. Após isso, o agente recebe um novo sinal de reforço r_{t+1} e uma nova indicação do estado do ambiente s_{t+1} em consequência de ação executada. Baseado nestes sinais o agente atualiza a estimativa do valor V ou Q do estado atual. Esse processo é repetido a cada interação com o ambiente. Na Figura 5.2 é ilustrada a interação entre o agente e o ambiente.

Um dos maiores desafios do AR é o compromisso entre a exploração: investigação de novas soluções no espaço de soluções e o reaproveitamento: melhoria de uma solução promissora já encontrada. Para obter uma grande recompensa, um agente precisa selecionar ações que já se mostraram promissoras no passado. Entretanto, para descobrir essas ações esse agente precisa explorar a execução das ainda não selecionadas. O agente precisa aprimorar as soluções já descobertas mas também precisa explorar novas soluções que levem a maiores recompensas. Neste modelo, nem a exploração nem o reaproveitamento podem ser tomados exclusivamente sem levar à falha do processo de aprendizado (Sutton and Barto, 1998).

No AR a ação do agente determina a recompensa imediata e também o novo estado do ambiente. Portanto, o agente precisa considerar o próximo

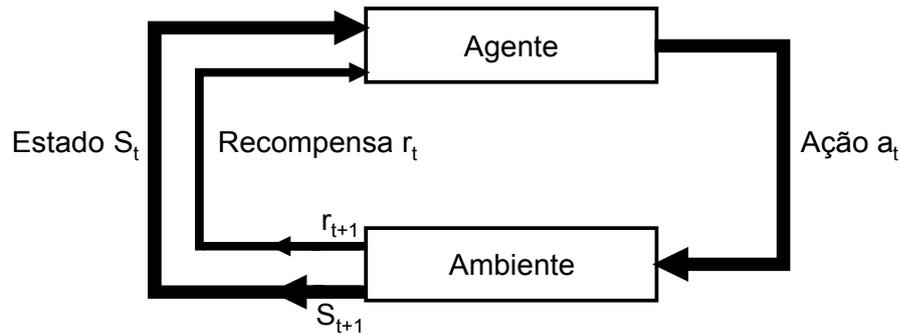


Figura 5.2: Interação entre o agente e o ambiente durante o processo de aprendizado por reforço. Após o agente perceber o estado atual do ambiente (S_t) ele seleciona uma ação (a_t) para ser executada e então verifica o novo estado do ambiente (S_{t+1}) e o reforço recebido como consequência da ação executada (r_t). (Sutton and Barto, 1998).

estado do ambiente e a recompensa imediata para a seleção de uma ação no estado atual. Em um processo de longo prazo, o agente precisa considerar as recompensas futuras a serem recebidas pela seleção de uma seqüência de ações até atingir um estado final que fornecerá um grande valor de reforço. Portanto, o agente precisa aprender por meio de recompensas futuras, sendo capaz de selecionar uma ação atual que leve a uma seqüência de estados e ações futuras que maximizem o total das recompensas a serem obtidas. Esse processo de seleção de ações que maximizem a soma dos sinais de reforço a serem obtidos a longo prazo geralmente é realizado por meio de tentativa e erro (Kaelbling et al., 1996) (Sutton and Barto, 1998).

Aprendizagem por Diferença Temporal

Os métodos de aprendizagem baseados em diferença temporal (TD) realizam predições a longo prazo sobre sistemas dinâmicos. Esses métodos são considerados promissores para a resolução de problemas de controle (Kaelbling et al., 1996) (Sutton and Barto, 1998). Eles possuem características comuns a outros métodos de aprendizagem por reforço: Monte Carlo e programação dinâmica. Métodos TD são capazes de aprender diretamente por meio da experiência, sem necessitar de um modelo do ambiente, como o método Monte Carlo. Métodos TD também são capazes de atualizar suas avaliações utilizando estimativas anteriormente aprendidas, sem esperar por um resultado ao final de uma seqüência de ações realizadas, como na programação dinâmica (Sutton and Barto, 1998). Os métodos TD esperam somente uma transição de estado para atualizar $V(s_t)$:

$$\Delta V(s_t) = \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)] \quad (5.1)$$

O ajuste da avaliação do estado s_t é baseado no retorno imediato obtido r_{t+1} e na estimativa do próximo estado $V(s_{t+1})$. Deste modo, o valor da avaliação para os métodos de diferença temporal é $[r_{t+1} + \gamma V(s_{t+1})]$, que constitui uma previsão para o objetivo da aprendizagem desta estratégia (Sutton and Barto, 1998).

Segundo Sutton (Sutton and Barto, 1998), existem dois tipos de frequência para atualização das estimativas da função avaliação, caracterizando os sistemas como problemas de um ou de múltiplos passos, sendo que para problemas reais os sistemas de múltiplos passos predominam. Em sistema de um passo, toda a informação sobre a predição da avaliação realizada é obtida a cada passo de tempo. Em sistema de múltiplos passos, o agente pode acessar somente a informação parcial sobre a avaliação desejada, sendo sua totalidade revelada somente após vários instantes de tempo. Dessa forma, os métodos de um passo envolvem a utilização do valor de avaliação do estado (ou do par estado-ação) considerando um único sucessor e os métodos de múltiplos passos se baseiam na distribuição completa dos valores de avaliação de todos os possíveis estados sucessores. A seguir são apresentados os principais algoritmos de aprendizado por diferença temporal.

Aprendizado-Q

O algoritmo do Aprendizado-Q (Watkins, 1989) é um dos métodos de aprendizado por diferença temporal mais simples de ser implementado. Este algoritmo foi utilizado para implementar o mecanismo de aprendizagem e emissão de respostas em uma das versões da arquitetura proposta e foi empregado por já ter sido utilizado com sucesso pelo grupo de pesquisa. Em sua forma mais simples, a regra de atualização de 1 passo é dada pela Equação 5.2:

$$\Delta Q(s_t, a_t) = \alpha [r_{t+1} + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \quad (5.2)$$

na qual α é a taxa de aprendizagem ($0 < \alpha < 1$) e γ é o parâmetro de desconto ($0 < \gamma < 1$).

Ao atualizar a função de avaliação Q utilizando o máximo valor $Q(s_{t+1}, a_{t+1})$, essa função de avaliação se aproxima da função da avaliação ótima Q^* , independente da política seguida (Watkins, 1989). A política, porém, ainda possui um efeito na aprendizagem, no sentido de definir quais pares estado-ação serão visitados e atualizados. A forma geral desse método é apresentado no Algoritmo 1 (Sutton and Barto, 1998).

A seguir são introduzidos os conceitos sobre aprendizado por reforço em domínios relacionais.

Algoritmo 1 Aprendizado-Q.

Requer: um conjunto de estados ($S = \{s_1, s_2, \dots, s_n\}$);

Um conjunto de ações ($A = \{a_1, a_2, \dots, a_n\}$);

Uma tabela representando a função ($Q = \{(s_i, a_i), \dots, (s_{i+n}, a_{i+n})\}$).

Inicie $Q(s, a)$ arbitrariamente

para cada tentativa **faça**

Obtenha o estado atual do ambiente

para cada passo em uma tentativa **faça**

Selecione uma ação a_t segundo a política π

Execute a ação a_t segundo a política π

Obtenha o retorno r_{t+1}

Obtenha o estado s_{t+1}

Atualize o valor da avaliação $Q(s_t, a_t)$ (Equação 5.2)

Faça $s_t = s_{t+1}$

fim para

fim para

5.3.2 Aprendizado por Reforço Relacional

O Aprendizado por Reforço Relacional (ARR) é a técnica que inspirou a representação do conhecimento da arquitetura proposta. Esta técnica foi selecionada como ponto de partida para a exploração das novas representações implementadas e propostas neste trabalho.

O AR oferece uma estrutura geral e diversos métodos para se construir agentes inteligentes que aperfeiçoam seu comportamento durante as interações com o ambiente. Entretanto, as representações convencionais utilizadas pelos métodos de RL, como as tabelas Q, dificultam sua aplicação em domínios reais com grandes espaços de busca, uma vez que o algoritmo de aprendizado pode levar um longo tempo para convergir (Otterlo, 2005). A utilização de uma representação tabular para a aproximação da função Q ou função V só é possível para tarefas com números pequenos de estados e ações.

Para que seja possível a execução do Q-learning em grandes ambientes, faz-se necessária a utilização de uma função de generalização que represente a *função-Q*. Esta generalização pode ser efetuada por um algoritmo de regressão que utiliza os exemplos (estado, ação e valor-Q) encontrados durante a exploração do ambiente. Então, um algoritmo regressivo constrói uma aproximação da função-Q, que aproxima as predições dos valores-Q para todo par (estado, ação), mesmo que esses nunca sejam encontrados durante a exploração. O uso de regressão para Q-learning não apenas reduz a quantidade de memória e o tempo necessário para a convergência para uma política ótima, mas também permite ao agente fazer predições da qualidade de pares (estado, ação) não vistos (Dzeroski et al., 2001) (Driessens, 2004) (Otterlo, 2005).

O ARR (Dzeroski et al., 2001) é uma combinação de AR e Programação

Lógica Indutiva (Mitchell, 1997b). Ele combina um algoritmo de AR padrão (como o *Q-Learning*), uma representação relacional de primeira ordem e algoritmos de regressão relacional que efetuam a generalização da função-*Q*. Esta generalização pode ser efetuada por algoritmos como o *TILDE-RT* (Blockeel et al., 1998). O *TILDE-RT* é um algoritmo para induzir árvores de regressão relacional e é uma extensão da técnica *C4.5* (Mitchell, 1997b). No ARR, o *TILDE-RT* é utilizado como uma ferramenta de representação da função-*Q* na forma de uma árvore de regressão de primeira ordem, denominada *árvore-Q* (Dzeroski et al., 2001) (Driessens, 2004) (Otterlo, 2005). O caminho desde a raiz até um nó folha contém diversas cláusulas relacionais que, quando percorridas, formam basicamente questões similares às questões relacionais em Prolog (Otterlo, 2005).

A forma de trabalhar o ARR é muito similar ao AR, exceto pela representação relacional usada para estados e ações. Pode-se definir RRL como (Driessens, 2004):

Dado

- um conjunto de possíveis estados S (representado em formato relacional),
- um conjunto de possíveis ações A (representado em formato relacional),
- uma função de transição desconhecida $\delta: S \times A \rightarrow S$ (essa função pode ser não determinística)
- uma função de recompensa $r: S \times A \rightarrow \mathbb{R}$.
- conhecimento prévio sobre o ambiente.

Encontre uma política de selecionar ações $\pi^*: S \rightarrow A$ que maximiza o valor da função $V^\pi(s_t)$ para todo estado $s_t \in S$.

Durante um episódio, as ações são tomadas de acordo com uma política atual, de acordo com a *árvore-Q* atual. Todos os novos pares de estado-ação encontrados são armazenados, enquanto os valores dos pares já encontrados são atualizados de acordo com o algoritmo de aprendizagem (*Q-Learning*). Depois de cada episódio, o *TILDE-RT* é usado para induzir uma árvore de regressão relacional sobre o conjunto de exemplos (estado, ação, valor-*Q*). (Otterlo, 2005).

Entretanto, a indução de uma nova *árvore-Q* após cada episódio de aprendizado pode ser um problema para domínios complexos com grandes espaços de busca (Driessens, 2004) (Otterlo, 2005). Para resolver este problema, foi proposto um algoritmo incremental denominado TG (Driessens, 2004).

O algoritmo TG (Driessens, 2004) é uma combinação do algoritmo de *TILDE-RT*, que constrói árvores de regressão e classificação de primeira ordem, e do algoritmo *Q* (Chapman and Kaelbling, 1991), que utiliza diversos valores estatísticos relativos ao desempenho de cada possível extensão em cada folha da árvore para construí-la de forma incremental. As árvores de regressão relacional empregadas pelo algoritmo TG utilizam conjunções de literais de primeira ordem como testes nos nós internos. O teste que corresponde a um certo nó folha é a conjunção dos testes que aparecem no caminho da raiz da árvore até o nó folha correspondente. O algoritmo TG emprega um operador de refinamento definido pelo usuário para gerar os testes de primeira ordem que podem ser usados para substituir um nó folha por um novo nó interno e então criar um novo nó folha. As estatísticas armazenadas pelo algoritmo TG em cada folha da árvore consistem no número de exemplos classificados positivamente ou negativamente por cada possível teste e a soma dos valores-*Q*, além do quadrado dos valores-*Q* em cada um destes casos. Isto permite o uso de um meta-teste para decidir qual teste será empregado na busca dos valores-*Q*.

Na Figura 5.3, é apresentado um exemplo de um episódio de aprendizado do ARR aplicado ao clássico problema do mundo dos blocos (Dzeroski, 2003).

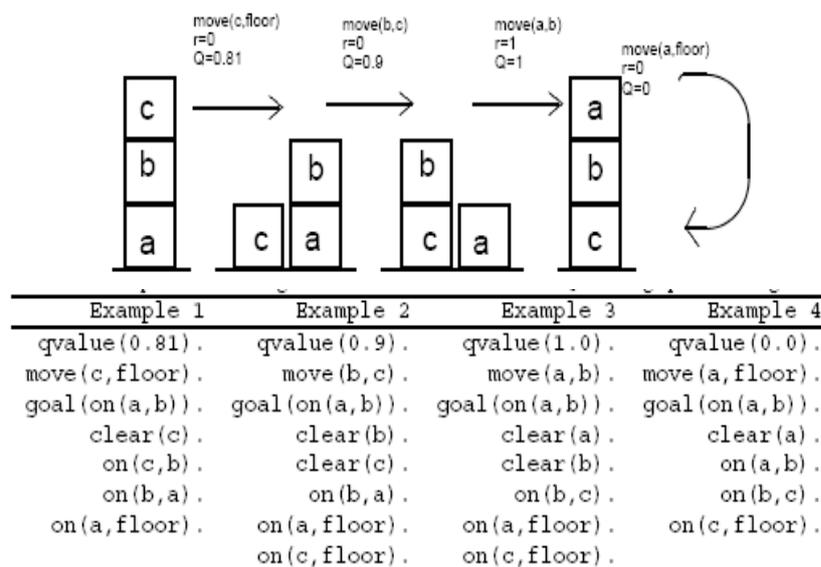


Figura 5.3: Episódio de aprendizado do ARR. Na metade superior pode-se ver a evolução dos estados do ambiente e na metade inferior pode-se ver as correspondente cláusulas Prolog da representação do ambiente.

A seguir é apresentada a Rede Neural ART 2.

5.3.3 Rede Neural ART 2

A Rede Neural ART 2 foi utilizada neste trabalho para implementar o mecanismo de regras de comportamento (memória de regras) e o mecanismo de aprendizado por tutelagem. Esta rede neural foi selecionada neste trabalho de pesquisa por já ter sido utilizada com sucesso por este pesquisador, em trabalhos prévios contendo estruturas similares às desenvolvidas no presente trabalho (Policastro et al., 2003).

As Redes Neurais baseadas no modelo ART 2 pertencem a uma classe de Redes Neurais que empregam aprendizado não supervisionado e que suportam o aprendizado contínuo, respondendo a mudanças dos padrões de entrada. Esse modelo de Rede Neural aceita entradas com valores contínuos ou binários e é composta por três camadas ou estágios de processamento (Carpenter and Grossberg, 1987).

A camada F_0 executa o pré-processamento e a normalização do padrão de entrada. Para fazer isto, esta camada utiliza uma função não-linear dependente do ruído das informações processadas. Esta função pode ser sigmoide ou linear threshold (Carpenter and Grossberg, 1987).

A camada F_1 processa o padrão filtrado (pela camada F_0) e compara com os protótipos dos agrupamentos (armazenados na camada F_2), por meio de um conjunto de pesos W (pesos *bottom-up*). Dado um padrão de entrada pré-processado, uma unidade de processamento em F_2 é selecionada. Essa unidade de processamento representa uma categoria (protótipo de um agrupamento). Uma vez selecionada uma unidade de processamento, ela envia seu protótipo para a camada F_1 por meio de um conjunto de pesos T (pesos *top-down*). A rede ART 2 efetua a comparação entre o protótipo do agrupamento selecionado e o padrão de entrada pré-processado. Se eles não são semelhantes o suficiente (dado um parâmetro de vigilância), o subsistema de orientação ativa um sinal de *reset* e desabilita a unidade de processamento selecionada em F_2 . Como resultado, um novo ciclo de busca é iniciado. Se o protótipo do agrupamento selecionado e o padrão de entrada são semelhantes, os pesos do agrupamento selecionado são atualizados (Carpenter and Grossberg, 1987).

A função da camada F_2 é efetuar a busca por um agrupamento (unidade de processamento) que seja mais similar ao padrão de entrada pré-processado. Esta busca é executada em paralelo. Todas as unidades entram em competição e a de maior ativação é a vencedora. Se a camada F_2 recebe um sinal de *reset* do subsistema de orientação, a unidade vencedora da competição é desabilitada e ela não pode mais entrar em competição nas fases de busca subsequentes, para o padrão de entrada atual. As unidades de processamento restantes competem novamente, e o ciclo se repete até uma unidade ser encontrada para acomodar o novo padrão de entrada, ou uma nova unidade ser

criada pela rede (Carpenter and Grossberg, 1987).

Externamente às camadas de processamento da rede (F_0 , F_1 , e F_2), a rede ART 2 possui um subsistema de orientação que estabelece o grau de similaridade entre os protótipos dos agrupamentos e o padrão de entrada pré-processado. A condição de *reset* é dada por um parâmetro de vigilância cujo valor deve estar entre 0 e 1 ($0 < \rho < 1$) (Carpenter and Grossberg, 1987). A arquitetura da rede ART 2 é ilustrada na Figura 5.4.

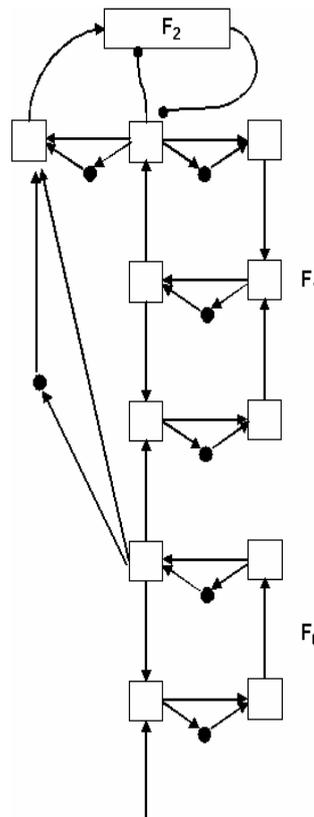


Figura 5.4: Arquitetura da rede ART2. As setas indicam a propagação das informações entre os nós de processamento das camadas F_0 , F_1 e F_2 (Carpenter and Grossberg, 1987).

A seguir é apresentada a técnica de aprendizagem K-Vizinhos mais Próximos (Knn).

5.3.4 K-Vizinhos mais Próximos

A técnica K-Vizinhos mais Próximos (Knn) foi utilizada neste trabalho para implementar a busca por regras no mecanismo de regras de comportamento.

A técnica *K-Vizinhos mais Próximos* é o mais básico método de aprendizado pertencente a uma família de métodos conhecidos como baseados em instância (Mitchell, 1997a). Os métodos de aprendizado baseados em instância são abordagens conceitualmente simples. O aprendizado nesses métodos

consiste no simples armazenamento das instâncias de treinamento em uma memória. Quando uma nova instância é apresentada ao sistema, um conjunto de instâncias similares é recuperado da memória do sistema e utilizado para classificar a nova instância.

A versão básica do algoritmo *K-Vizinhos mais Próximos* assume que todas as instâncias correspondem a pontos em um espaço n - dimensional em \mathbb{R}^n . Os vizinhos mais próximos de uma instância são definidos em termos de uma distância padrão, geralmente a distância *Euclidiana*. Mais precisamente, seja uma instância arbitrária descrita por um vetor $\langle a_1(x), a_2(x), \dots, a_n(x) \rangle$, no qual $a_r(x)$ denota o valor do r -ésimo atributo da instância x . Então, a diferença entre duas instâncias x_i e x_j é definida como $d(x_i, x_j)$ e é dada pela Equação 5.3 (Mitchell, 1997a):

$$d(x_i, x_j) \equiv \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2} \quad (5.3)$$

na qual n é o número de atributos presentes nas instâncias do conjunto de dados, $a_r(x_i)$ é o valor do r -ésimo atributo da instância x_i e $a_r(x_j)$ é o valor do r -ésimo atributo da instância x_j . No algoritmo *K-Vizinhos mais Próximos*, a função objetivo (que produz os valores do atributo classe) pode assumir tanto valores discretos como valores contínuos (Mitchell, 1997a). Para problemas com função objetivo de valores discretos dados por uma função $f : \mathbb{R}^n \rightarrow V$, na qual V é o conjunto finito $\{\nu_1, \nu_2, \dots, \nu_s\}$ de valores que o atributo classe pode assumir, a versão básica do algoritmo *K-Vizinhos mais Próximos* é ilustrada pelo Algoritmo 2. Nesse Algoritmo, o valor $h(I_c)$ retornado é o valor mais freqüente encontrado entre os k vizinhos mais próximos da instância a ser classificada (I_c). O algoritmo *K-Vizinhos mais Próximos* pode facilmente ser adaptado para problemas com função objetivo de valores contínuos (Mitchell, 1997a).

A seguir são apresentadas as bibliotecas de software empregadas para a construção dos sistemas de apoio da arquitetura proposta.

5.4 Sistemas de Apoio

Nesta Seção são apresentadas as bibliotecas de software empregadas para a construção do sistema de visão e do sistema de reconhecimento de fala, utilizados pela arquitetura proposta, assim como introduzidas as técnicas empregadas por estas bibliotecas.

Algoritmo 2 Versão básica do algoritmo *K-Vizinhos mais Próximos* para problemas com função objetivo de valores discretos

Requer: Um conjunto de instâncias de treinamento rotuladas ($I = \{I_1, I_2, \dots, I_n\}$);

Uma instância de consulta a ser classificada (I_c);

O número de vizinhos mais próximos a ser considerado (k);

A função de distância entre instâncias (d).

Dado o conjunto I

Dada a instância I_c a ser classificada

Determine o conjunto $\hat{I} = \{\hat{I}_1, \hat{I}_2, \dots, \hat{I}_k\}$ de k instâncias mais próximas à instância de consulta I_c

$$\text{Faça: } h(I_c) = \underset{\nu \in V}{\text{argmax}} \sum_{i=1}^k \delta(\nu, f(\hat{I}_i))$$

na qual $\delta(a, b) = 1$ se $a = b$ e $\delta(a, b) = 0$ caso contrário

retorne $h(I_c)$

fim função

5.4.1 Visão Computacional

O sistema de visão construído durante o desenvolvimento deste trabalho integra a detecção de objetos salientes e faces humanas. Esse sistema cria um mapa de ativação que pode ser então utilizado pela arquitetura para simular as preferências visuais dos recém nascidos. Para a construção deste sistema, foram empregadas duas bibliotecas de processamento de imagem e visão computacional, disponíveis para trabalhos acadêmicos e de pesquisa.

A biblioteca Lti-lib (Lti-Lib, 2003) é uma biblioteca totalmente orientada a objetos com algoritmos e estrutura de dados freqüentemente usados no processamento de imagens e em visão computacional. Esta biblioteca foi desenvolvida no Instituto de Ciência da Computação da *Universidade Tecnológica de Aachen* e é o resultado de muitos anos de pesquisa para solução de problemas de visão computacional, tais como reconhecimento de objetos, visão para robôs móveis e reconhecimento de gestos.

A biblioteca Watson (Morency, 2007) é uma biblioteca de visão computacional desenvolvida no *Massachusetts Institute Technology* (MIT) como resultado das pesquisas na área de visão computacional, interação homem-máquina e aprendizado de máquina. A biblioteca é o resultado das pesquisas em adaptive view-base appearance models (AVAM) ou modelo adaptativo de visão computacional baseado em aparência (Morency et al., 2003). O Adaptive View-Base Appearance Models (AVAM) é uma representação flexível e compacta de objetos e é usada para rastrear e estimar as mudanças de pose de uma face humana (Morency and Darrell, 2002).

Atenção Visual

O módulo de detecção de objetos salientes foi construído empregando as funções de saliência da Lti-Lib, baseadas no trabalho de Itti, uma das abordagens mais utilizadas em atenção visual bottom-up (Itti et al., 1998). Esta abordagem é baseada em um mapa bidimensional que representa a saliência visual dos objetos. O mapa de saliência é formado pela composição de diversos mapas de características extraídas da imagem. Cada mapa de característica apresenta uma propriedade elementar da imagem como cor, intensidade e orientação. Essas características são conhecidas como características visuais primitivas. O método pode ser dividido nas seguintes etapas: extração de características, filtragem linear, cálculo das diferenças centro-vizinhas, soma de mapas de características (combinação linear) e seleção de regiões salientes.

Para gerar um mapa de saliência, os três mapas de características visuais primitivas são extraídas (cor, intensidade e orientação). O mapa de cor é gerado empregando os três canais as cores de entrada r , g e b do *frame* da imagem recebida a partir câmera do robô. Esses canais normalizados são então utilizados para produzir quatro canais de cores denominados r' , g' , b' e y' , dados pelas Equações (5.4), (5.5), (5.6) e (5.7).

$$r' = r_n - (g_n + b_n)/2 \quad (5.4)$$

$$g' = r_n - (r_n + b_n)/2 \quad (5.5)$$

$$b' = r_n - (r_n + g_n)/2 \quad (5.6)$$

$$y' = \frac{r_n + g_n}{2} - b_n - \|r_n - g_n\| \quad (5.7)$$

O mapa de intensidades é dado pela Equação (5.8)

$$I = \frac{(r + g + b)}{3} \quad (5.8)$$

que define a imagem em tons de cinza. Para cada canal de cor e para o mapa de intensidades, são criadas Pirâmides Gaussianas de nove níveis: $R(\sigma)$, $G(\sigma)$, $B(\sigma)$, $Y(\sigma)$, no qual $\sigma \in 0, \dots, 8$. As Pirâmides Gaussianas são geradas utilizando um algoritmo proposto por Burt e Adelson (Burt and Adelson, 1983).

As informações de orientação local são obtidas pela aplicação de um algoritmo proposto por Freeman e Adelson (Freeman and Adelson, 1991), que constrói uma estrutura de representação piramidal direcional da imagem. Os canais de cores e o mapa de intensidades são submetidos a um processo de fil-

tagem linear. Nesta decomposição linear, realizada pela estrutura piramidal, uma imagem é subdividida em um conjunto de sub-bandas localizadas em escala e orientação. A representação piramidal é usada para a obtenção de amostras da imagem somente com os detalhes relevantes ao processamento. Os mapas de características são obtidos por meio da diferença entre canais de cores em diferentes escalas, este processo é conhecido como diferença centro-vizinhança. Nesta subtração de imagens, o centro é um pixel da imagem em uma escala $c \in \{2, 3, 4\}$ e a vizinhança é o pixel correspondente de outra imagem em uma escala $s = c + \delta$ com $\delta \in \{3, 4\}$ da pirâmide. Então, computam-se seis mapas da característica para cada tipo de característica (nas escalas 2-5, 2-6, 3-6, 3-7, 4-7, 4-8). Desta forma, obtém-se sete tipos de características: uma para intensidade da imagem (On-Off), duas que codificam as diferenças Vermelho-Verde e Azul-Amarelo, e mais quatro para contraste local da orientação (0° , 45° , 90° , 135°) (Itti et al., 1998) (Breazeal et al., 2001).

Detecção de Face e Pose

O módulo de detecção e rastreamento de faces foi construído empregando as funções de detecção da Watson, baseadas nos trabalhos de Morency (Morency et al., 2003) (Morency et al., 2003), sobre a detecção de uma face e da pose (ângulos *pan* e *tilt*) desta face.

Dada uma imagem arbitrária ou uma seqüência de imagens, o objetivo da detecção de face é determinar se há ou não qualquer face humana na imagem e, se alguma estiver presente, retornar sua localização e extensão espacial. Uma abordagem recentemente muito empregada para a detecção de faces é a baseada em aparências (Morency et al., 2003) (Morency, 2007). Esta abordagem não utiliza nenhum conhecimento a priori sobre o objeto ou características a ser detectada. Nesta classe de algoritmos, surge os conceitos de aprendizado e treinamento, uma vez que as informações necessárias para realizar a tarefa de detecção são retiradas do próprio conjunto de imagens e a técnica de análise de componentes principais (PCA) para gerar o modelo da face a ser utilizado para a detecção de uma face.

Uma vez detectada uma face em uma imagem, pode-se proceder a detecção da pose da mesma. Esta detecção pode ser dividida em duas partes (Morency et al., 2003):

1. Criação do modelo a priori dado um *frame* inicial;
2. Cálculo das mudanças de poses usando um Modelo Baseado em Aparência.

A criação do modelo a priori, na biblioteca Watson, foi implementada empregando o método PCA para 28 diferentes poses de 14 diferentes pessoas.

Inicialmente o sistema procura uma reconstrução com menor erro para a pose inicial e a partir dos coeficientes de correspondência, dados pela técnicas PCA, reconstrói a imagem para todas as outras poses. Isso equivale a procurar a melhor representação da imagem para todas as poses. O propósito de reconstruir um modelo *a priori* é generalizar um grupo de poses que é utilizado no cálculo da pose relativa entre dois *frames*. Dado um *frame* próximo e a pose do modelo P , é possível reconstruir todas as outras possíveis poses nos *frames* subsequentes. O modelo *a priori* construído é então utilizado para estimar a pose absoluta de um novo *frame* por meio de duas etapas:

- Cálculo da pose relativa entre o novo *frame* e cada pose do modelo *a priori* usando o algoritmo iterativo AVAM.
- Integrar a pose mensurada (pose relativa) usando o *Filtro de Kalman* para produzir a pose absoluta.

O modelo AVAM proposto por Morency (Morency et al., 2003) mantém as diversas poses de um objeto em *frames-chaves*. O método armazena um *frame* atual, um *frame* anterior e todos os *frames-chaves*. O modelo pode ser atualizado e ajustado, isto é, *frames-chaves* podem ser modificados, adicionados ou removidos durante a busca. o modelo AVAM consiste em uma coleção de poses armazenadas em *frames-chaves* adquiridos durante a detecção e cálculo das poses. Para cada *frame-chave* o modelo armazena a seguinte informação:

$$M_s = \{I_s, Z_s, x_s\} \quad (5.9)$$

Na qual, I_s e Z_s são as imagens de intensidade e de profundidade associadas ao *frame chave* s . O modelo, portanto, pode ser definido pelo conjunto $\{M_1, M_2, \dots, M_k\}$ onde k é o número de *frames-chaves*.

A seguir é apresentada a biblioteca de software empregada para a construção do sistema de reconhecimento de fala utilizado pela arquitetura proposta.

5.4.2 Reconhecimento de Fala

O sistema de reconhecimento de fala construído durante o desenvolvimento deste trabalho é capaz de reconhecer frases completas em português. Este sistema é parte integrante da interface multimodal utilizada pela arquitetura proposta. Para a construção deste sistema, foi empregada a biblioteca *Nuance* de reconhecimento de fala (Nuance, 2001).

A biblioteca Nuance é baseada em uma arquitetura cliente/servidor, composta por um módulo cliente de reconhecimento responsável pela aquisição

de fala (incluindo o pré-processamento da mesma), por um servidor de reconhecimento responsável pelo reconhecimento de fala, por um gerenciador de recursos responsável pela distribuição eficiente de tarefas de reconhecimento.

O processo de reconhecimento da biblioteca Nuance pode ser dividido em diversas fases. Como parte do processo, o servidor de reconhecimento recebe o sinal da fala como entrada e utiliza três componentes para executar o reconhecimento (Nuance, 2001):

- Modelos acústicos: modelos acústicos são providos pela biblioteca Nuance e utilizados para o reconhecimento fonético;
- Arquivos de dicionário: os arquivos de dicionário contêm as descrições das pronúncias fonéticas das palavras de uma gramática;
- Gramática de reconhecimento: uma gramática de reconhecimento define um conjunto de expressões vocais que podem ser reconhecidas e as interpretações para estas expressões vocais.

Antes da fala ser enviada ao servidor de reconhecimento, o cliente de reconhecimento efetua um pré-processamento os dados de áudio para otimizar o reconhecimento. Uma das principais operações de pré-processamento é a distinção do ruído de fundo da própria expressão vocal. Este processo é chamado de *Endpointing* e permite à biblioteca Nuance filtrar um número variado de ruídos de fundo (Nuance, 2001).

Após receber a fala e durante a fase de busca, o servidor analisa as características da mesma para produzir uma transcrição da expressão vocal para texto. A procura é definida por um conjunto de possibilidades especificadas na gramática atual. Examinando estas possibilidades, o servidor de reconhecimento utiliza uma hierarquia de mecanismos de busca que permitem a ele selecionar as hipóteses mais prováveis a partir de um conjunto de possíveis hipóteses (Nuance, 2001):

- No nível mais baixo, são reconhecidos fonemas individuais utilizando-se os modelos acústicos especificados. Um fonema é um som que distingue um significado em um idioma. Cada fonema pode ser estendido por múltiplos *frames*. Os modelos de fonema podem ser dependentes do contexto, o que significa que eles podem depender dos fonemas precedentes e seguintes.
- Seqüências de fonemas compõem palavras. O servidor de reconhecimento utiliza dicionários juntamente com regras conversão de texto para sons para mapear seqüências de fonema a palavras.

- Palavras são combinadas para a montagem de frases ou orações. As gramáticas definidas para uma aplicação determinam o conjunto de seqüências de palavras que o servidor de reconhecimento pode entender.

A fase de análise acústica-fonética do processo de reconhecimento provê um mapeamento probabilístico da forma da onda da expressão vocal para um conjunto de possíveis fonemas. Uma vez que palavras e orações são construídas a partir de modelos fonéticos, é importante que o processo acústico seja preciso. Para tanto, a biblioteca nuance emprega modelos de *Markov* ou *hidden Markov models (HMM)* (Lawrence, 1989) como modelos acústicos para mapear a expressão vocal a uma seqüência de fonemas (Nuance, 2001).

Durante o mapeamento da forma da onda da expressão vocal em uma seqüência de fonemas e orações, o sistema executa duas tarefas simultâneas como parte do processo de reconhecimento, segmentação e classificação. A análise de segmentação determina em que posição de uma fala estão as palavras e em que posição nestas palavras estão os fonemas. A classificação determina, para cada segmento, qual é o fonema e qual palavra foi ouvida, data uma seqüência de fonemas (Nuance, 2001).

Durante o reconhecimento dos fonemas, a máquina de reconhecimento da biblioteca Nuance utiliza métodos de poda para reduzir o espaço de busca. Existem dois métodos de poda utilizados. A poda baseada em contagens de probabilidade, no qual o servidor de reconhecimento computa a probabilidade de todas as possíveis hipóteses na gramática e escolhe uma com a maior probabilidade dada o modelo acústico (HMM). A poda fonética, no qual o servidor executa computações adicionais baseadas no último fonema analisado em um determinado momento durante o reconhecimento. A poda fonética provê uma avaliação independente que aumenta a probabilidade de o servidor de reconhecimento tomar a decisão certa sobre manter ou podar uma hipótese.

5.5 Considerações Finais

Nesse Capítulo foi apresentada a proposta deste trabalho de pesquisa, com a descrição da arquitetura desenvolvida. Adicionalmente, foram apresentadas as técnicas empregadas para o desenvolvimento desta pesquisa.

As técnicas e algoritmos apresentados neste Capítulo foram definidas após as pesquisas bibliográficas realizadas. Estas técnicas foram empregadas para a implementação das estruturas e mecanismos necessários para a realização do trabalho proposto.

O AR é um modelo promissor para a simulação do aprendizado de agentes por interações com o ambiente e tem sido aplicado a diversos problemas nos últimos anos. Esse modelo de aprendizado se mostra capaz de simular

um importante mecanismo de aprendizagem e modificação de comportamento baseado na teoria da Análise do Comportamento. Portanto, este foi selecionada como a técnica base para a construção do algoritmo de aprendizado da arquitetura proposta.

Para superar os problemas encontrados em grandes espaços de estado pelas implementações de AR convencionais e explorar novas representações de conhecimento, a arquitetura proposta neste trabalho emprega uma representação híbrida do conhecimento baseada em ARR e Redes Neurais.

Adicionalmente, para o desenvolvimento dos sistemas de visão e reconhecimento de voz que integram a arquitetura proposta, foram empregadas as técnicas de visão computacional e reconhecimento de fala apresentadas neste Capítulo, por serem técnicas muito utilizadas na literatura e pela facilidade de desenvolvimento dos sistemas pela utilização das bibliotecas de software disponíveis.

Uma vez apresentado todo o embasamento teórico e as técnicas empregadas para o desenvolvimento deste projeto de pesquisa, no próximo Capítulo é apresentada a arquitetura proposta em detalhes. Adicionalmente, são apresentadas os experimentos realizados e os resultados obtidos, assim como uma discussão detalhada sobre os mesmos.

Resultados

6.1 *Considerações Iniciais*

Neste Capítulo são apresentados os experimentos executados durante o desenvolvimento deste trabalho, assim como os resultados obtidos e algumas discussões específicas sobre os resultados obtidos em cada estudo. As discussões gerais sobre a arquitetura e sobre esta pesquisa são feitas no próximo Capítulo (ver Considerações Finais).

Este Capítulo está organizado da seguinte maneira. Na Seção 6.2, é discutida brevemente a metodologia de projeto empregada para o desenvolvimento deste trabalho. Na Seção 6.3, são introduzidos os domínios dos problemas nos quais a arquitetura foi aplicada para a realização dos experimentos e avaliações da mesma. Na Seção 6.4, são discutidas as métricas e procedimentos para os experimentos realizados com a arquitetura proposta. Na Seção 6.5, é apresentada a cabeça robótica interativa empregada neste trabalho. Na Seção 6.6, é detalhada a arquitetura proposta, apresentando todos os módulos desenvolvidos. Na Seção 6.7, é brevemente apresentado o simulador de uma Caixa de Skinner, desenvolvido durante este projeto de pesquisa, bem como são apresentados os resultados sobre os estudos de simulação de um rato em uma Caixa de Skinner. Na Seção 6.8, é descrito o simulador de interações sociais, também desenvolvido durante esta pesquisa, assim como os resultados obtidos sobre os experimentos de simulação do aprendizado da atenção compartilhada. Na Seção 6.9, são apresentados os resultados obtidos sobre os estudos realizados empregando esta cabeça robótica. Finalmente, na Seção 6.10, são feitas as considerações finais deste Capítulo.

6.2 Metodologia de Projeto

A metodologia empregada para o desenvolvimento desta pesquisa é semelhante à metodologia apresentada na Seção 2.2 do Capítulo 2. Após as pesquisas bibliográficas necessárias ao embasamento teórico, foram realizadas algumas atividades de planejamento e definição da arquitetura proposta neste trabalho. Após estas atividades, a primeira versão da arquitetura proposta foi modelada. Esta versão foi implementada e testada em uma simulação de um *rato* em uma *Caixa de Skinner*, um experimento clássico da Psicologia (Catania, 1999) (Serio et al., 2004). Após esta validação, foram verificadas diversas oportunidades de melhorias nos mecanismos e estruturas da arquitetura, de forma a torná-la mais plausível e adequada às teorias nas quais ela foi inspirada. Nesta etapa do projeto, foi realizado também um estudo sobre a convergência do algoritmo de aprendizado da arquitetura. A partir deste estudo e da validação, iniciou-se um novo ciclo de planejamento, definição e modelagem para o aperfeiçoamento dos mecanismos existentes e implementação de novos mecanismos na arquitetura. Entre outras, uma importante modificação executada foi a reformulação da regra de aprendizagem empregada pela arquitetura para possibilitar a sua convergência para uma solução ótima. Após isso, a arquitetura foi novamente validada empregando-se o simulador da *Caixa de Skinner* para determinar a eficácia das mudanças realizadas e determinação da capacidade de aprendizado da arquitetura.

Após esta nova validação, foi realizada uma pequena modificação do algoritmo de recuperação das regras de comportamento para otimizar o tempo de recuperação. Posteriormente, foram realizados diversos experimentos para a validação da arquitetura em um problema de aplicação real e não trivial, o aprendizado da atenção compartilhada (Dube et al., 2004) (Kaplan and Hafner, 2004). Para tanto, foi desenvolvido um simulador de interações sociais (detalhado na Seção 6.8.1 do Capítulo 6), capaz de simular os movimentos necessários ao robô e ao ser humano, além de fornecer estímulos apropriados do ambiente. Após este estudo, novos mecanismos foram incorporados à arquitetura para possibilitar a generalização do conhecimento adquirido durante o processo de aprendizado. Então, foi realizado um novo estudo para a comparação do desempenho desta nova versão da arquitetura com a versão anterior, além da comparação destas com uma versão da arquitetura empregando um algoritmo Q-Learning clássico como algoritmo de aprendizado.

Finalmente, após este estudo, a arquitetura recebeu um módulo de visão computacional e um módulo de reconhecimento de fala, além de um sistema motor e um novo mecanismo de aprendizado. Após estas últimas extensões, foi realizado um estudo empregando a versão final da arquitetura para con-

trolar uma cabeça robótica interativa em um experimento de aprendizado da atenção compartilhada em um ambiente social real. Adicionalmente, foi realizado um estudo de aprendizado por tutela para avaliar o novo mecanismo de aprendizagem incorporado à arquitetura. Todos estes estudos e validações são apresentados e detalhados no Capítulo 6. O processo de projeto empregado neste trabalho pode ser visto na Figura 6.1.



Figura 6.1: Ciclo de projeto empregado neste trabalho. Após cada ciclo, uma validação foi realizada por meio de diversos experimentos. Então, uma nova funcionalidade foi incorporada à arquitetura, de forma que esta tivesse todas as funcionalidades requeridas, de maneira incremental.

6.3 Domínios de Aplicação

Para a realização dos estudos e avaliações previstos neste projeto de pesquisa, a arquitetura proposta foi aplicada a dois domínios de problema com relevância para a Psicologia e para a área da robótica sociável: *caixa de Skinner* e *Atenção Compartilhada*.

6.3.1 Caixa de Skinner

Uma caixa de Skinner contém tipicamente uma ou mais barras que um animal pode pressionar, um ou mais estímulos luminosos e um ou mais lugares nos quais podem ser entregues estímulos reforçadores, como comida ou água. A ação de pressionar uma barra pode ser detectada e registrada e, o estabelecimento de uma contingência entre estas ações, os estímulos antecedentes

presentes no ambiente e a entrega de estímulos reforçadores pode ser programada automaticamente. Também é possível produzir estímulos aversivos ou castigos como choque elétrico pelo chão da caixa (Serio et al., 2004).

Se a caixa é programada de forma que a pressão na barra resulta na entrega de uma quantidade de alimento ou água somente quando uma luz de controle está acesa, então um animal poderá aprender a pressionar a barra no contexto correto se deixado interagir por um longo tempo com os dispositivos da caixa. Entretanto, pode-se modelar o comportamento do rato inicialmente de forma que ele aprenda a comer de uma maneira mais eficiente e efetiva. Por exemplo, ao invés de recompensar o rato somente quando ele pressionar a barra com a luz de controle acesa, pode-se inicialmente recompensá-lo quando ele se aproxima da região na qual se encontra a barra. Após uma história de reforçamento, o animal aprenderá a se aproximar da barra sempre que quiser uma porção de comida ou água. Depois, pode-se iniciar uma fase na qual o reforço somente é fornecido se sua cabeça estiver apontando na direção da barra e, então, somente quando o animal pressionar ou esbarrar na barra. Desta forma, após uma história de reforçamento, se o rato explorar sozinho a caixa de Skinner, é provável que este aprenderá a pressionar a barra somente quando a luz de controle estiver acesa. Na Figura 6.2 é ilustrada uma caixa de Skinner com um rato em seu interior.

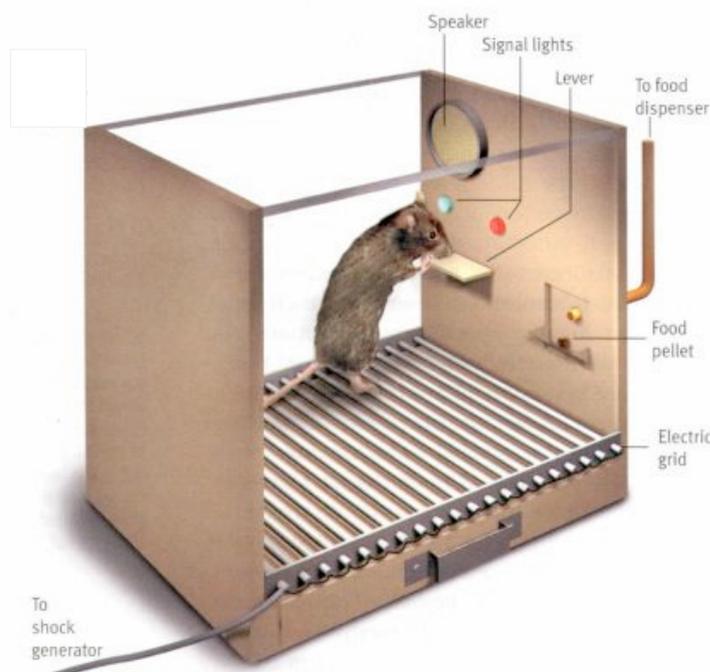


Figura 6.2: Caixa de Skinner com um rato. Após o rato pressionar a barra com a luz de controle acesa, uma porção de alimento ou água pode ser entregue como recompensa para ação do rato.

6.3.2 Atenção Compartilhada

A atenção compartilhada é definida na literatura como o “a capacidade de utilizar gestos e contato ocular para coordenar a atenção de outro agente, e compartilhar objetos ou eventos de interesse no ambiente” (Dube et al., 2004) (Kaplan and Hafner, 2004). A atenção visual compartilhada é associada à situação na qual dois agentes estão olhando um para o outro, um agente direciona o seu olhar para um objeto presente no ambiente e, então, o segundo agente segue o olhar ao objeto correto. Esta habilidade torna possível o aprendizado do que é importante no ambiente (Deák and Triesch, 2005).

Dube e seus colegas (Dube et al., 2004) apresentaram uma análise para explicar as origens de atenção compartilhada. Esta análise recorre a situações nas quais uma criança inicia a seção de atenção compartilhada, direcionando a atenção de um adulto para um objeto no ambiente e obtendo a atenção deste adulto durante a interação com o objeto. Porém, esta análise pode ser estendida para explicar sessões de atenção compartilhada iniciadas também por um adulto. A análise caracteriza o início de um evento interessante em um contexto do ambiente que inclui a presença de um adulto familiar como Operador Motivacional. Um Operador Motivacional é um evento que muda o estado do ambiente e reforça uma resposta de um indivíduo. Um comportamento torna-se mais freqüente se o Operador Motivacional estabelece um reforço ou inibe um castigo, ou fica menos freqüente se o Operador Motivacional inibe um reforço ou estabelece um castigo. O efeito da mudança de comportamento de um Operador Motivacional pode ser visto como a mudança positiva ou negativa da freqüência do comportamento relevante aos eventos conseqüentes e pode depender da presença de estímulos discriminativos apropriados no ambiente.

Em uma sessão de atenção compartilhada, um evento interessante estabelece no ambiente a capacidade de reforço de uma classe de estímulos denominados estímulos de atenção de um adulto. Estes estímulos são percepções visuais e auditivas indicativas da atenção dos adultos a um objeto ou evento de interesse. As respostas de um adulto torna-se efetivas como reforçador condicional após uma história de reforçamento: os estímulos discriminam que o adulto reagirá a um evento interessante e a reação do adulto é relacionada a um aumento da probabilidade de acesso a reforçadores. Então, o comportamento da criança de seguir o olhar do adulto é resultado de uma história de reforçamento na qual a atenção do adulto torna-se o estímulo reforçador. Ainda, a atenção dos adultos age como uma ponte em uma cadeia de comportamentos (Dube et al., 2004).

6.4 Análise dos Resultados

A análise do desempenho da arquitetura proposta foi baseada na quantificação do aprendizado do domínio de aplicação (Caixa de Skinner e Atenção Compartilhada). Os experimentos foram compostos por uma fase de aprendizado com duração diferente para cada estudo realizado durante o desenvolvimento deste trabalho. Para os experimentos simulados da Caixa de Skinner, a fase de aprendizado foi executada com duração de 4.000 unidades de tempo, ou 4.000 ciclos. Para os experimentos simulados da Atenção compartilhada, a fase de aprendizado foi executada com duração de 10.000 unidades de tempo. Para os experimentos com o robô real, a fase de aprendizado foi executada com duração de 1.000 unidades de tempo.

Para os experimentos sobre a atenção compartilhada, a capacidade de aprendizado da arquitetura foi analisada pela observação da interação do robô com o humano e o ambiente, e pela computação de uma métrica denominada *correct gaze index* ou *índice de olhar correto* (CGI). Esta métrica foi adaptada à partir dos trabalhos apresentado por Whalen (Whalen and Schreibman, 2003) para a estrutura dos experimentos realizados e é definida como a frequência de direcionamento da atenção para o local correto, o qual o humano está olhando. Esta métrica é dada pela Equação (6.1):

$$CGI = \frac{\#deslocamentos \ a \ partir \ do \ humano \ para \ a \ localizacao \ correta}{\#deslocamentos \ a \ partir \ do \ humano \ para \ qualquer \ localizacao} \quad (6.1)$$

Para os experimentos sobre a Caixa de Skinner, a capacidades de aprendizado da arquitetura foi analisada pela observação da interação do rato com o ambiente e pela computação de uma métrica denominada *press bar index* ou *índice pressão a barra* (PBI), proposta deste trabalho. Esta métrica é definida como a frequência de ações de pressionar a barra de alimentação da caixa de Skinner com a luz de controle acesa. Esta métrica é dada pela Equação (6.2):

$$PBI = \frac{\#Pressao \ da \ barra \ com \ a \ luz \ acesa}{\#Pressao \ da \ barra} \quad (6.2)$$

Para quantificar o aprendizado da arquitetura durante os experimentos, a fase de aprendizado era interrompida em pontos específicos, a cada 500 unidades de tempo (para os experimentos do o robô a fase de aprendizado era interrompida a cada 100 unidades de tempo). Então, uma fase de validação do conhecimento adquirido era iniciada para avaliar o comportamento da arquitetura. Esta avaliação foi executada por 20 corridas de 500 unidades de tempo ou 500 ciclos de interação (para os experimentos do robô eram executadas 20 corridas de 100 unidades de tempo). Para cada corrida, foi computado o valor

do CGI ou PBI, dado pelas Equações (6.1) (6.2) respectivamente. Depois das 20 corridas, a média e desvio padrão das 20 medidas foram calculados, dados respectivamente pelas Equações (6.3) e (6.4). Depois das 20 corridas da fase de avaliação, a fase de aprendizado foi retomada do ponto no qual esta havia sido interrompida.

$$\overline{CGI} = \frac{1}{k} \sum_{i=1}^k CGI_i \quad e \quad \overline{PBI} = \frac{1}{k} \sum_{i=1}^k PBI_i \quad (6.3)$$

$$d = \sqrt{var(A)} \quad (6.4)$$

na qual:

$$var(A) = \frac{1}{k} \left[\frac{1}{k-1} \sum_{i=1}^k (CGI - \overline{CGI})^2 \right] \quad ou \quad var(A) = \frac{1}{k} \left[\frac{1}{k-1} \sum_{i=1}^k (PBI - \overline{PBI})^2 \right]$$

Todo o processo de experimentação foi repetido com a variação dos parâmetros empregados pela arquitetura, para se determinar o melhor conjunto de parâmetros de aprendizado e para se verificar a influência da variação destes parâmetros no desempenho da arquitetura. Para a comparação dos algoritmos com diferentes parâmetros de aprendizado, foi aplicado um teste estatístico para comprovar o desempenho dos resultados dos algoritmos, denominado *teste t* para procedimentos bilaterais. Esse teste estatístico permite comprovar com um elevado grau de confiança se o desempenho de um algoritmo é realmente superior ao desempenho de outro. Basicamente, as seguintes etapas devem ser seguidas para a realização desse teste (Moses, 1986) (Mitchell, 1997a), considerando dois algoritmos *A* e *B* utilizando diferentes conjuntos de parâmetros de treinamento:

1. Estabelecer a hipótese Inicial nula $H_0 : \bar{e}_A = \bar{e}_B$ e a hipótese alternativa $H_1 : \bar{e}_A \neq \bar{e}_B$;

na qual:

$$\bar{e} = 1 - CGI \quad ou \quad \bar{e} = 1 - PBI$$

2. Considerando n_A e n_B o número de padrões utilizados pelo algoritmo *A* e *B*, respectivamente, deve-se determinar um valor comum para n , dado pela Equação 6.5:

$$n = \frac{d_A^2 + d_B^2}{\frac{d_A^2}{n_A} + \frac{d_B^2}{n_B}} \quad (6.5)$$

onde d_A^2 é o desvio padrão do algoritmo *A* e d_B^2 é o desvio padrão do algoritmo *B*.

3. Determinar os limites da região de aceitação $RA : (-k_{\frac{\alpha}{2}}, k_{\frac{\alpha}{2}})$ na tabela de distribuição de *Student t* com base no valor do nível de significância (α) e no valor de n ;
4. Realizar os teste nos algoritmos e calcular a média do erro absoluto e o desvio padrão;
5. Calcular o valor do teste estatístico (t_{calc}), dado pela Equação 6.6:

$$t_{calc} = \frac{\bar{e}_A - \bar{e}_B}{\sqrt{\frac{d_A^2}{n} + \frac{d_B^2}{n}}} \quad (6.6)$$

onde \bar{e}_A é a média do erro do algoritmo A , \bar{e}_B é a média do erro do algoritmo B , d_A^2 é o desvio padrão do algoritmo A e d_B^2 é o desvio padrão do algoritmo B .

Se o valor de t_{calc} estiver dentro da RA , então os algoritmos possuem desempenhos equivalentes. Se o valor de t_{calc} for positivo e estiver fora da RA , então o algoritmo B é melhor que o algoritmo A com $X\%$ de confiança. Se o valor de t_{calc} for negativo e estiver fora da RA , então o algoritmo A é melhor que o algoritmo B com $X\%$ de confiança.

6.5 Cabeça Robótica Interativa

Os experimentos de interações sociais reais da arquitetura foram realizados empregando-se uma cabeça robótica interativa conectada por cabo a um computador. Esta cabeça robótica é composta por 5 motores (servos), uma câmera digital colorida, um módulo multimídia e um módulo controlador. Este robô é apresentado na Figura 6.3.

Esta cabeça robótica possui um *kit de desenvolvimento de software* (SDK) composto por um controle *Active X* contendo uma série de funções da interface de programação *API* do controlador do robô, que pode ser adicionado em programas desenvolvidos em *Visual C++*. Para o desenvolvimento de um sistema de controle, deve-se então adicionar este *Active X* em um projeto de software, no ambiente de programação *IDE do Visual C++* e acessar as rotinas de controle disponibilizadas por este componente de controle.

Existem diversas funções de controle disponibilizadas no SDK do robô e estas funções são agrupadas nas categorias de Sensores Periféricos, Controle Motor, Controle Multimídia e Eventos. Entretanto, devido a versão do robô somente as funções de controle dos servos foram empregadas nesta pesquisa. A seguir são apresentadas as funções de controle empregadas no desenvolvimento do sistema de controle do robô.



Figura 6.3: Cabeça robótica interativa juntamente com o sistema de controle.

- *EnableServo (short channel)* – liga o canal (servo motor) especificado de controle.
- *DisableServo (short channel)* – desliga o canal (servo motor) especificado de controle.
- *ServoTimeCtr(short channel, short cmdValue, short timePeriod)* – envia um comando de controle de movimento ao canal do controle do servo motor especificado (*channel*). Este comando especifica que o servo motor deve atingir uma determinada posição (*cmdValue*) em um determinado tempo (*timePeriod*).
- *void ServoNonTimeCtr(short channel, short cmdValue)* – envia um comando de controle de movimento ao canal do controle servo do motor especificado (*channel*). Este comando especifica que o servo motor deve atingir uma determinada posição (*cmdValue*), sem determinar tempo de ação.
- *ServoTimeCtrAll(short cmd1, short cmd2, short cmd3, short cmd4, short cmd5, short cmd6, short timePeriod)* – envia simultaneamente um comando de controle de movimento para todos os canais do controle dos servo motores. Este comando especifica que os servo motores devem atingir determinadas posições (*cmd1...cmd6*) em um determinado tempo (*timePeriod*).

- *ServoNonTimeCtrAll* (*short cmd1*, *short cmd2*, *short cmd3*, *short cmd4*, *short cmd5*, *short cmd6*) – envia simultaneamente um comando de controle de movimento para todos os canais do controle dos servo motores. Este comando especifica que os servo motores devem atingir determinadas posições (*cmd1...cmd6*), sem determinar tempo de ação.

As funções acima descritas, disponibilizadas no SDK do robô, foram empregadas para o desenvolvimento de diversas rotinas motoras que foram utilizadas pelo sistema de controle e pela arquitetura proposta durante os experimentos de interação com o robô em um ambiente social real e controlado.

6.6 *Arquitetura Proposta*

Nesta seção, é apresentada e detalhada a arquitetura robótica proposta, composta por mecanismos e estruturas evidenciadas na Análise do Comportamento. Esta arquitetura integra mecanismos de percepção e aprendizagem, além de uma representação alternativa do conhecimento adquirido e mantido pela mesma. Adicionalmente, esta arquitetura integra um sistema de visão computacional, um sistema de reconhecimento de fala, um sistema motor desenvolvido para a cabeça robótica interativa utilizada neste trabalho, e os módulos básicos de controle inspirados na Análise do Comportamento. Ainda, esta arquitetura integra um mecanismo capaz de simular o aprendizado de conceitos sobre objetos do mundo real apresentados ao robô, por meio da tutela de um ser humano. Na Figura 6.4 é apresentada a arquitetura desenvolvida.

Em cada iteração do ciclo de controle, o sistema codifica os estímulos visuais e auditivos dos sistemas de visão e reconhecimento de fala, respectivamente. Então, o sistema de controle envia os estímulos codificados para a arquitetura robótica que, por sua vez, executa uma iteração de aprendizado e indica uma resposta a ser emitida como consequência desta iteração. Posteriormente, o sistema de controle chama a rotina motora apropriada e emite a resposta designada pela arquitetura robótica, completando um ciclo de iteração com o ambiente.

A seguir, são apresentadas as estruturas e métodos da arquitetura proposta.

6.6.1 *Sistema de Visão Computacional*

O sistema de visão é composto por um módulo de reconhecimento de face capaz de estimar a pose da cabeça de um ser humano, baseado em modelos adaptativos de visão computacional baseados em aparência (Morency et al.,

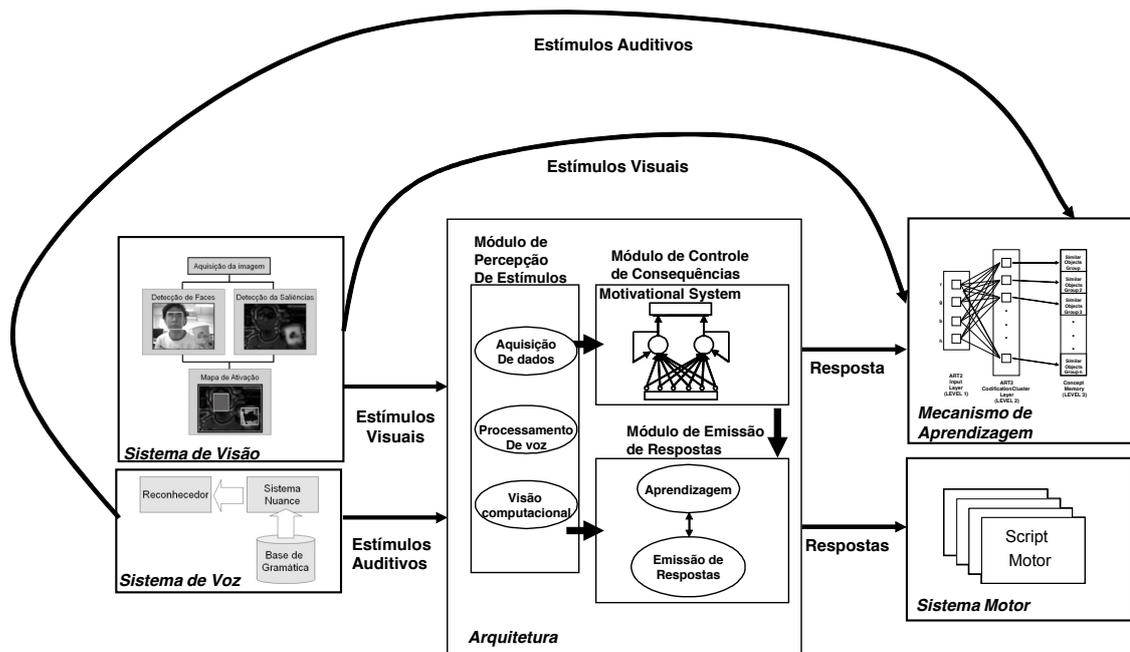


Figura 6.4: Arquitetura proposta. As setas indicam o fluxo de informações entre os módulos do sistema de controle. Este sistema emprega os diversos módulos desenvolvidos durante este projeto de pesquisa: sistema de visão, sistema de voz, mecanismo de aprendizagem por tutelação, sistema motor e arquitetura robótica.

2003). O sistema de visão também é composto por um módulo de detecção de objetos, baseado em modelos de saliência e atenção visual (Itti et al., 1998).

O sistema de visão é baseado no trabalho apresentado por Breazeal e Scassellati (Breazeal and Scassellati, 1999) e é capaz de simular algumas preferências visuais de crianças entre os 6 e os 18 meses de idade, como cores salientes e faces humanas. A implementação do sistema de visão é baseado em mapas de características processados para cada percepção (cores e faces). O resultado do processo deste sistema consiste em um mapa de ativação que pode ser usado pelos outros módulos da arquitetura de controle para controlar o comportamento do robô.

O mapa de cores é baseado no trabalho de busca e atenção visual apresentado por Itti e seus colegas em (Itti et al., 1998). Este processo utiliza um mecanismo biologicamente inspirado de atenção visual para criar um mapa de características que representa a *saliência visual* de uma cena. Esta saliência visual é formada pela composição de vários mapas de características da imagem, como apresentado na Seção 5.4 do Capítulo 5. Este mapa de saliência foi desenvolvido utilizando-se as funções de saliência da Lti-Lib (Lti-Lib, 2003).

O mapa de faces é baseado nos trabalhos apresentados por Morency (Morency et al., 2003) (Morency, 2007), sobre face e detecção de poses. A detecção

das faces é executada empregando-se uma abordagem de modelos adaptativos baseados em aparência (Morency et al., 2003) (Morency, 2007), como apresentado na Seção 5.4 do Capítulo 5. Este mapa de face foi desenvolvido empregando-se as funções de detecção de face e pose da Watson (Morency, 2007).

Depois de processar estes mapas de características (saliência e face), estes são utilizados para construir um mapa de ativação que representa os estímulos detectados pelo sistema de visão. O mapa de saliência é processado empregando-se um limiar de saliência e um limiar de raio mínimo de área para selecionar uma região de interesse a partir das saliências apontadas no mapa. Este processo é baseado no trabalho apresentado por Rodrigues (Rodrigues and Gomes, 2002). Então, um processamento baseado em histograma de cores é executado para se obter os valores mais freqüentes dos canais r , g , b (do espaço de cores RGB) e h (do espaço de cores HSI), dentro da região de interesse. O mapa de face é processado, empregando-se as funções da biblioteca Watson, para se obter a posição (x e y) das faces detectadas e a pose destas faces (ângulos pan e tilt). Posteriormente, os atributos de cores de objetos salientes e pose das faces são codificados e propagados para outros módulos da arquitetura para controlar o comportamento do robô. Na Figura 6.5, é ilustrada a arquitetura geral do sistema de visão computacional desenvolvido.



Figura 6.5: Sistema de visão computacional. *Detecção de Faces* ilustra o mapa de faces e *Detecção de Saliências* ilustra o mapa de cores. *Mapa de Ativação* ilustra o mapa de ativação resultante da combinação dos mapas processados pelo sistema de visão.

6.6.2 Sistema de Voz

O sistema de voz é composto por um sistema de reconhecimento de fala e por um sistema de vocalização. O sistema de reconhecimento de fala é capaz de reconhecer a linguagem natural falada no idioma Português Brasileiro e é baseado no Sistema Nuance (Nuance, 2001).

Este sistema contém uma máquina de reconhecimento de fala e uma base de conhecimento gramatical. Adicionalmente, existe um módulo reconhecedor implementado em linguagem Java, por permitir um fácil interfaceamento com o sistema *NuanceTM*. Este módulo reconhecedor recebe a codificação da fala, relativa à gramática configurada na base de conhecimento e a envia para o sistema de controle da cabeça interativa por meio de uma porta *socket*.

O sistema de vocalização, nesta versão do sistema de voz, utiliza diversos arquivos *.wav* gravados com palavras individuais previstas no vocabulário o robô. Desta forma, o sistema de voz pode montar uma frase desejada para que o robô a vocalize, pela união destas palavras individuais. Este sistema habilita sessões de conversações curtas com o robô, suficientes para os propósitos dos experimentos realizados neste trabalho de pesquisa. Na Figura 6.6, é ilustrada a arquitetura geral do sistema de reconhecimento de fala.

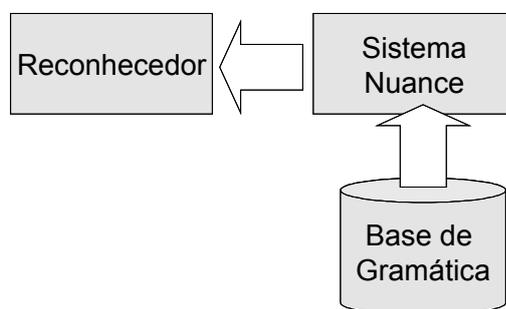


Figura 6.6: Sistema de reconhecimento de fala. O sistema *NuanceTM* (*Sistema Nuance*) utiliza a base de conhecimento gramatical (*Base de Gramática*) para reconhecer uma fala detectada no ambiente. Este módulo é responsável por enviar a codificação de uma fala reconhecida ao módulo reconhecedor implementado em *JavaTM*, que por sua vez disponibiliza esta codificação para o sistema de controle da cabeça robótica por meio de uma porta de *socket*.

6.6.3 Sistema Motor

O sistema motor da cabeça robótica interativa foi implementado empregando-se as funções disponibilizadas no SDK do robô. Este sistema motor possui scripts com diversos comandos temporizados de motor, que permitem ao robô emitir seis comportamentos diferentes:

1. Procurar por um ser humano, no qual o robô parte da sua posição atual e posiciona a sua cabeça de forma centralizada e ereta, para que este enxergue uma área imediatamente à sua frente, na linha do horizonte.
2. Procurar por um objeto a frente e abaixo, no qual o robô parte da sua posição atual e posiciona a sua cabeça de forma centralizada e abaixada, para que este enxergue uma área imediatamente a sua frente, mas abaixo do horizonte.
3. Procurar por um objeto a esquerda, no qual o robô parte da sua posição atual e posiciona a sua cabeça de forma ereta, mas virada para a sua esquerda, para que este enxergue uma área à esquerda, na linha do horizonte.
4. Procurar por um objeto a direita, no qual o robô parte da sua posição atual e posiciona a sua cabeça de forma ereta, mas virada para a sua direita, para que este enxergue uma área à direita, na linha do horizonte.
5. Procurar por um objeto a esquerda e abaixo, no qual o robô parte da sua posição atual e posiciona a sua cabeça virada para a sua esquerda e abaixada, para que este enxergue uma área à esquerda e abaixo do horizonte.
6. Procurar por um objeto a direita e abaixo, no qual o robô parte da sua posição atual e posiciona a sua cabeça virada para a sua direita e abaixada, para que este enxergue uma área à direita e abaixo do horizonte.

Estes seis scripts de comandos motores permitem a emissão dos comportamentos necessários aos experimentos reais de interação no contexto do aprendizado da atenção compartilhada e do aprendizado por tutelação.

6.6.4 Mecanismo de Aprendizagem por Tutelagem

O mecanismo de aprendizagem por tutelação proposto neste trabalho de pesquisa é capaz de associar estímulos visuais e auditivos para simular a aprendizagem de conceitos sobre objeto do mundo real por meio da tutelação de um ser humano.

Este mecanismo utiliza o sistema de visão para extrair as características visuais de um determinado objeto, composto pelos valores mais freqüentes dos canais r, g, e b (do espaço de cores RGB), e do canal h (do espaço de cores HSI). O mecanismo de aprendizagem também utiliza o sistema de voz para adquirir o nome falado do objeto apresentado ao robô. Estas características são então aprendidas e organizadas empregando uma rede neural do

tipo ART2 (Carpenter and Grossberg, 1987) e uma memória plana que armazena as características visuais juntamente com o nome falado do objeto para formar novos conceitos. O mecanismo de aprendizagem contém três níveis de organização de memória, como ilustrado na Figura 6.7:

1. O primeiro nível (NÍVEL 1 na Figura 6.7) é composto pela camada de entrada da rede neural ART2. Esta camada contém quatro nós de entrada, um para cada canal de cor (r, g, b, e h);
2. O segundo nível (NÍVEL 2 na Figura 6.7) é composto pela camada de saída da rede neural ART2. Esta camada cria e/ou indica agrupamentos de objetos com características semelhantes, possibilitando o reconhecimento e aprendizagem de conceito e a generalização do conhecimento;
3. O terceiro nível (NÍVEL 3 na Figura 6.7) é composto por uma memória plana que armazena as características visuais e auditivas dos objetos.

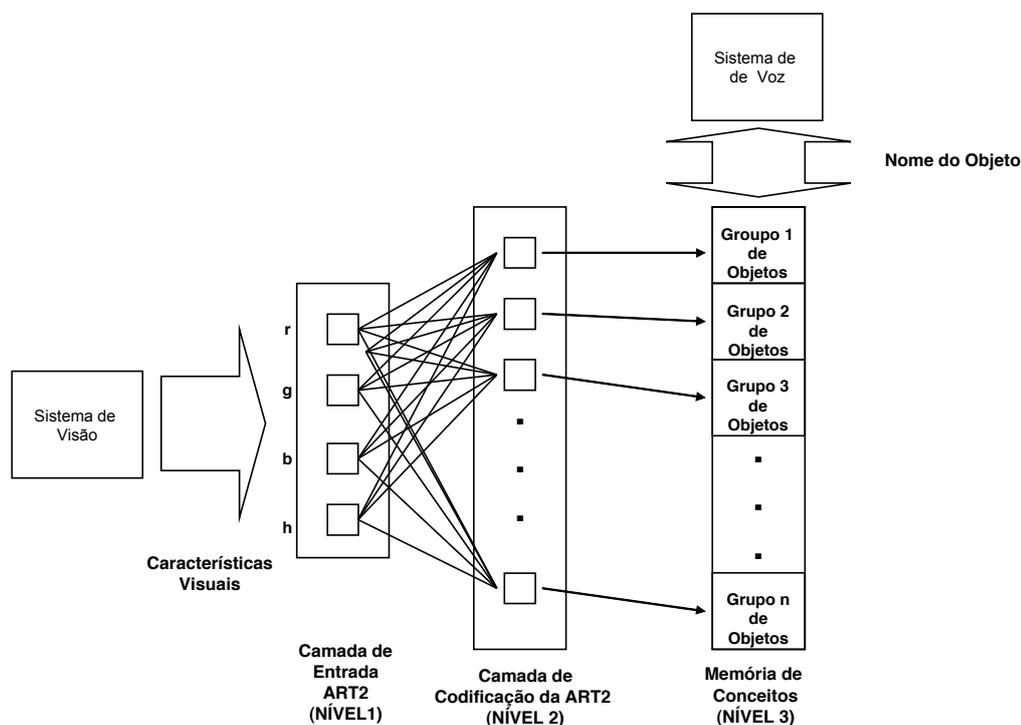


Figura 6.7: Arquitetura geral do mecanismo de aprendizagem. A camada de entrada da rede neural ART2 recebe as características de cor de um objeto (r, g, b, e h) do sistema de visão e indica a codificação de *camada de saída de ART2* do objeto. Então, para um objeto desconhecido, o mecanismo de aprendizagem obtém o significado do mesmo, do sistema de voz, e integra estas informações visuais e auditivas para formar um conceito novo na memória de conceitos.

O mecanismo de aprendizagem trabalha da seguinte maneira. Inicialmente, a memória de conceito está vazia. Quando um objeto é apresentado ao

robô, o sistema de visão codifica este objeto por seus valores mais frequentes dos canais r, g, b, e h. A camada de entrada da rede neural ART2 recebe as características de cor deste objeto (r, g, b, e h) do sistema de visão e indica que não há nenhum *cluster* ativo em sua camada de saída. Então, o mecanismo de aprendizagem entra em um estado de *desconhecimento* e ativa o sistema de voz para vocalizar que o objeto é desconhecido. Depois, o mecanismo aguarda o correto significado do objeto, por meio do sistema de reconhecimento de fala. Posteriormente, a rede neural ART2 é treinada no modo rápido (Carpenter and Grossberg, 1987) no qual os pesos das sinapses são atualizados durante o processo de ressonância da rede neural, tomando somente o tempo de uma apresentação do padrão de entrada. Como resultado, o novo conceito aprendido é armazenado na memória de conceitos.

A partir disso, quando são apresentados novos objetos ao robô, a camada de entrada da rede neural ART2 recebe as características de cor do objeto, por meio do sistema de visão, e indica o cluster de codificação, se há um, para o novo objeto. Então, o algoritmo de busca procura por objetos, dentro do cluster indicado, empregando uma métrica dado por: $m = ||h_n - h_r||$, na qual h_n é o valor do h do novo objeto, e h_r é o valor do h de um objeto armazenado na memória de conceitos. Se o algoritmo de busca encontrar algum objeto com métrica de similaridade abaixo de um limiar de confiança ϕ_c , o mecanismo de aprendizagem entra em um estado de *certeza* e ativa o sistema de voz para vocalizar o nome do objeto.

Se o algoritmo de busca encontrar somente objetos acima um limiar de conhecimento ϕ_k , o mecanismo de aprendizagem entra em um estado de *desconhecimento* e ativa o sistema de voz para vocalizar que o objeto é desconhecido. Depois, o mecanismo aguarda o significado correto do objeto, por meio do sistema de reconhecimento de fala. Então, a rede neural ART2 é treinada no modo rápido (Carpenter and Grossberg, 1987) no qual os pesos das sinapses são atualizados durante o processo de ressonância da rede neural, tomando somente o tempo de uma apresentação do padrão de entrada. Como resultado, o novo conceito aprendido é armazenado na memória de conceitos.

Se o algoritmo de busca encontrar somente objetos entre o limiar de confiança e o limiar de conhecimento, o mecanismo de aprendizagem entra em um estado de *incerteza* sobre o objeto e ativa o sistema de voz para vocalizar que o objeto é supostamente o objeto mais similar encontrado na memória de conceitos. Depois, o mecanismo aguarda a confirmação do significado ou o significado correto do objeto, por meio do sistema de reconhecimento de fala, e armazena o novo conceito aprendido na memória de conceitos caso o mecanismo tenha errado o seu *chute* sobre o nome do objeto.

6.6.5 Arquitetura básica

A arquitetura básica é composta por três módulos principais: percepção de estímulos, emissão de respostas e controle de conseqüências.

O módulo de percepção de estímulos pode empregar algoritmos de aquisição de dados, um sistema de visão computacional e um sistema de reconhecimento de fala, dependendo do domínio da aplicação para o qual se está desenvolvendo o robô. Este módulo é responsável por detectar o estado do ambiente e codificar este estado empregando uma representação apropriada.

O módulo de emissão de respostas é composto por um mecanismo de aprendizagem capaz de construir uma política não determinística para a emissão de respostas, isto é, quais respostas podem ser emitidas na presença de certos estímulos no ambiente.

O módulo de controle de conseqüências é composto por um *sistema motivacional* que simula as necessidades internas de um indivíduo e detecta os reforços recebido do ambiente. O sistema motivacional é formado por unidades de necessidade que são implementadas como um *perceptron* simples com conexões recorrentes (Haykin, 1999). Estas unidades de necessidade simulam a homeostase de um organismo vivo, provendo um mecanismo para simular estados de privação e saciação, além de um mecanismo para determinar reforços como conseqüências de respostas emitidas. Na Figura 6.8 é ilustrada a organização geral da arquitetura proposta e a interação entre os três módulos principais.

Representação do Conhecimento

A representação de conhecimento adotada para a arquitetura proposta é baseado em uma representação relacional de primeira ordem (Driessens, 2004) (Otterlo, 2005), possibilitando a representação e a manipulação de grandes espaços de busca e decisão de um modo econômico, em relação a um algoritmo de aprendizado por reforço clássico. A arquitetura codifica conhecimento como estímulos, fatos, respostas, regras de comportamento e regras de restrição.

Os *estímulos* codificam todos os estímulos detectados no ambiente e são representados como átomos ou objetos que possuem propriedades como cor, tamanho, forma, posição e pose (para faces). As propriedades dos estímulos são preenchidas pelo módulo de percepção de estímulos e servem para codificar o estado atual do ambiente. Por exemplo, pode-se definir *face* como um estímulo a ser detectado no ambiente, por meio de uma função do módulo de percepção de estímulos que preencherá a propriedade de cor com o valor de *cor de pele*, além de valores apropriados para as propriedades posição e

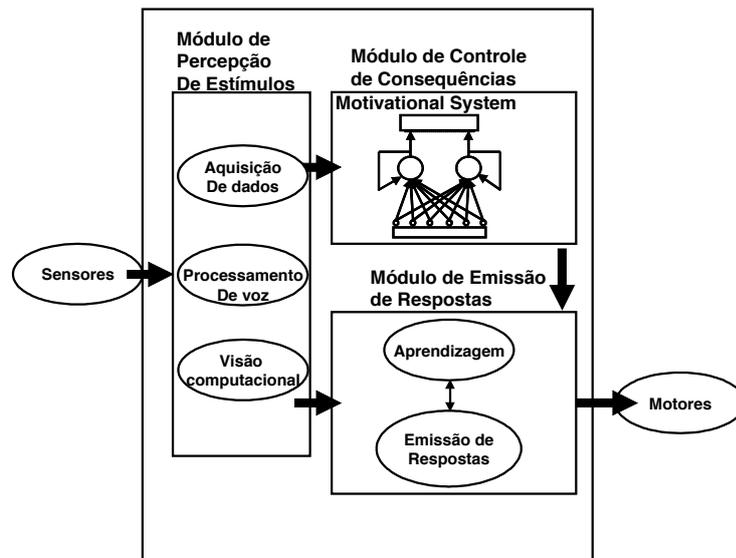


Figura 6.8: Organização geral da arquitetura proposta. As setas indicam o fluxo de informações entre os três módulos da arquitetura. Os círculos indicam os métodos e estruturas componente dos módulos. O módulo de percepção de estímulos codifica os estímulos detectados no ambiente. Esses estímulos são então utilizados pelos módulos de controle de consequências e de emissão de respostas para aprender a exibir comportamentos apropriados.

ângulos de pose da cabeça (*pan* e *tilt*). Nesta arquitetura, o estado do ambiente é codificado pelos seguintes predicados de percepção: *see(X)*, *hear(Y)*, *at(Z)* e *smell(W)*. Estes predicados qualificam todos os estímulos detectados no ambiente para construir uma representação do estado atual do mesmo.

Os *fatos* são conceitos declarados sobre determinados estímulos. Eles permitem a generalização e diferenciação de uma classe de estímulos a partir das propriedades dos mesmos. Por exemplo, pode-se declarar um estímulo *object* e declarar que qualquer objeto vermelho ou amarelo é um brinquedo escrevendo: *toy(object):-color(red)*; *toy(object):-color(yellow)*.

As respostas são codificadas na arquitetura como predicados de ação do robô. Estes predicados representam rotinas motoras pré-definidas que podem ser executados para emitir uma resposta selecionada pela arquitetura. Por exemplo, pode-se codificar as respostas *lookhuman()* e *looktoy()* e associá-las às funções motoras correspondentes para representar os comportamentos alguns inatos do robô.

Adicionalmente, a arquitetura proposta codifica o conhecimento na forma de regras de comportamento e regras de restrição. As regras de comportamento codificam o conhecimento sobre o comportamento apropriado aprendido pela arquitetura e possuem a seguinte forma geral:

$$\text{estímulos} \xrightarrow[\text{necessidades}]{Q_i} \text{resposta} \quad (6.7)$$

na qual a parte antecedente é formada por um conjunto de estímulos que representam um estado do ambiente e a parte conseqüente é a resposta a ser emitida pelo robô. O valor Q (Q_i) é o valor de aptidão da regra e é utilizado para calcular a probabilidade de execução da regra quando a parte antecedente for satisfeita, representando um integrante de uma função similar à Tabela Q de um algoritmos de AR clássico. O valor *necessidades* indica quais necessidades internas (unidades do sistema motivacional), se estas existirem para a regra, serão satisfeitas como conseqüência da execução de uma determinada regra. Ambos os valores são empregados pela seleção de resposta para selecionar a regra a ser emitida na presença de estímulos discriminativos.

As regras de restrição são empregadas para indicar quando certas respostas podem ou não podem ser emitidas e possuem a seguinte forma geral:

$$estimulos \rightarrow pode(resposta) \quad (6.8)$$

na qual a parte antecedente é um conjunto de estímulos que representa um estado do ambiente, e a parte conseqüente é a resposta que só pode ser emitida se a parte antecedente for satisfeita. Estas regras são empregadas pela arquitetura para restringir a emissão de algumas respostas.

Adicionalmente, a representação do conhecimento proposta neste trabalho emprega uma organização de memória, denominada *mecanismo de regras de comportamento* (MRC), que acelera o processo de recuperação das regras de comportamento por meio da divisão e redução do espaço de busca, permitindo também a generalização incremental do conhecimento adquirido. Nesta representação, as regras de comportamento aprendidas que compartilham estímulos similares são agrupadas em *clusters*. O MRC contém três níveis de organização de memória, como ilustrado na Figura 6.9:

- o primeiro nível (NÍVEL 1 na Figura 6.9) é composto pela camada de entrada de uma rede neural do tipo ART2. Esta camada de entrada contém um nó de entrada para cada estímulo que pode ser detectado e codificado a partir do ambiente.
- O segundo nível (NÍVEL 2 na Figura 6.9) é composto pela camada de saída da rede neural do tipo ART2, que cria e também indica agrupamentos das regras de comportamento semelhantes, reduzindo o espaço de busca e o tempo de recuperação das regras de comportamento. Adicionalmente, este nível da memória possibilita a generalização de conhecimento adquirido.
- O terceiro nível (NÍVEL 3 na Figura 6.9) é composto por uma memória plana simples que armazena as regras de comportamento agrupadas segundo os agrupamentos de semelhança do segundo nível da memória.

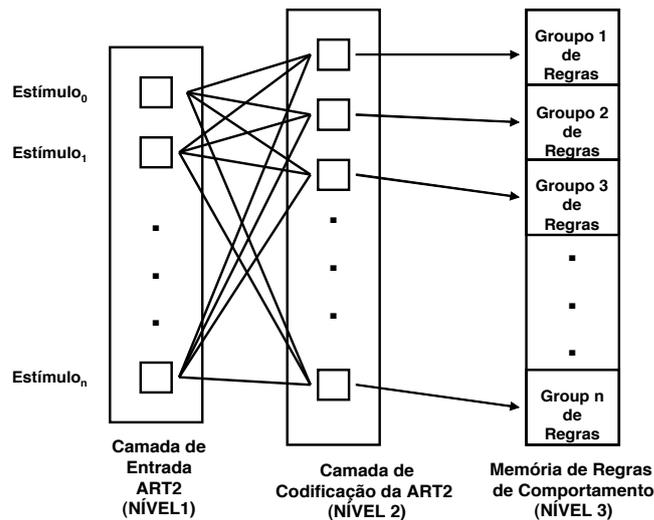


Figura 6.9: Organização geral do mecanismo de regras de comportamento. A camada de entrada da rede neural do tipo ART2 (NÍVEL 1) recebe os estímulos codificados do ambiente ($Estimulo_0..Estimulo_n$) e cria ou indica um agrupamento de regras de comportamento semelhantes (NÍVEL 2). Durante o processo de aprendizado, o mecanismo armazena a nova regra de comportamento na memória planta, em um grupo associado ao agrupamento indicado pela rede neural. Durante o processo de recuperação, um algoritmo de busca recupera o conjunto de regras de comportamento indicadas pelo *Cluster* e envia este conjunto de regras para o módulo emissão de respostas.

Quando a arquitetura robótica precisa aprender uma nova regra de comportamento, empregando o mecanismo de aprendizado do módulo de emissão de respostas, esta é enviada ao MRC. Então, este mecanismo extrai e pré-processa os estímulos codificados e armazenados na regra de comportamento, como parte antecedente da mesma, para formar um padrão de entrada a ser apresentado para a rede neural ART2. Então, a rede neural ART2 é treinada no modo rápido (Carpenter and Grossberg, 1987) no qual os pesos das sinapses são atualizados durante o processo de ressonância da rede neural, tomando somente o tempo de uma apresentação do padrão de entrada. Como resultado, a rede neural ART2 indica um agrupamento ao qual a nova regra de comportamento pertence. Então, o MRC armazena esta nova regra de comportamento na memória plana (NÍVEL 3) associada ao agrupamento apontado pela rede neural ART2 (NÍVEL 2).

Quando a arquitetura robótica precisa recuperar e utilizar um conjunto de regras de comportamento apropriadas para o contexto atual, o MRC pré-processa os estímulos detectados no ambiente, recebidos do módulo de percepção de estímulos, para formar um padrão entrada para a rede neural ART2. Então, este padrão de entrada é apresentado à rede neural, que indica o agrupamento de regras mais similar ao padrão de entrada. Posteriormente, o MRC seleciona o conjunto de regras de comportamento, pertencentes ao agrupamento

de semelhança indicado pela rede neural ART2. Então, um algoritmo de busca recupera todas as regras de comportamento, do conjunto selecionado, que possuem sua parte de antecedente satisfeita pela codificação atual do ambiente. Finalmente, estas regras recuperadas são enviadas ao módulo de emissão de respostas que executa a seleção final e a emissão de resposta mais apropriada.

Memória de Trabalho

A arquitetura proposta emprega uma memória de trabalho para trocar informação entre os seus três módulos principais. Esta memória é usada para manter informações sobre estímulos, últimas respostas emitidas, necessidades internas ativas e reforços detectados. Cada elemento inserido na memória de trabalho possui um contador que confere a noção de tempo. Quando um novo elemento é inserido na memória, seu contador de tempo é inicializado com valor igual a zero. Adicionalmente, este contador é incrementado de 1 sempre que novos elementos são inseridos subsequente na memória. Assim, os elementos persistem por vários passos de tempo na memória. Este mecanismo é empregado para controlar a cronologia de fatos e eventos e determinar os três termos de uma contingência.

Na Figura 6.10 é ilustrado o conteúdo da memória de trabalho, durante a simulação de uma interação social. Esta simulação mostra o momento no qual o robô detecta e estabelece contato visual com um ser humano para estabelecer uma interação social.

Memória de trabalho	
4	at(environment) & see(object)
3	need(socialize)
2	searchhuman()
1	at(environment) & see(frontal (face))
0	get(attention)

Figura 6.10: Fotografia da memória de trabalho. Os números representam o mecanismo de contador de tempo. Os elementos representam o estado do ambiente, as respostas emitidas, as necessidades ativas e os reforços recebidos.

Sistema Motivacional

O comportamento operante de um organismo depende da motivação destes e do valor da recompensa ou punição como conseqüência de um comportamento. Um sistema motivacional artificial pode possibilitar a um robô intera-

gir pró-ativamente com o ambiente, influenciando seus comportamentos para satisfazer as necessidades artificiais internas. Neste contexto, o módulo de controle de conseqüências emprega um sistema motivacional que simula as necessidades internas de um indivíduo. O sistema motivacional proposto foi adaptado a partir dos trabalhos apresentados por Breazeal (Breazeal, 2002) e Gadanho (Gadanho and Hallan, 2002). Entretanto, a arquitetura proposta no presente trabalho de pesquisa difere ligeiramente daqueles trabalhos, apresentando arquitetura similar a uma rede neural recorrente, formando uma arquitetura mais simples e possibilitado a rápida configuração do sistema motivacional para os diversos domínios de aplicação. O sistema motivacional proposto utiliza uma ou mais unidades de necessidade implementadas como um perceptron simples com conexões recorrentes (Haykin, 1999). A ativação de uma unidade de necessidade é determinada por:

$$u = \left(\sum_{j=1}^n w_j \times i_j \right) + w_r \times i_r + b \quad (6.9)$$

na qual i_j é o sinal de entrada que representa a codificação de um estímulo detectado no ambiente, i_r é o sinal da conexão recorrente, w_j é o peso de cada conexão do sinal de entrada, w_r é o peso da conexão recorrente e b é o *bias* da unidade. Todos os pesos e *bias* são empiricamente definidos de acordo com a necessidade a ser simulada. A saída de uma unidade de necessidade é determinada por uma função sigmóide, dada por:

$$y = \frac{1}{1 + e^{-(u+\delta)}} \quad (6.10)$$

na qual u é o valor de ativação e δ é a inclinação da função sigmóide. Adicionalmente, o sistema motivacional possui um mediador que decide o valor de saída do sistema motivacional, verificando o maior valor de ativação entre as diversas unidades de necessidade, empregando um esquema competitivo e um limiar de ativação, ou seja, o valor de saída do sistema motivacional será o maior valor de ativação entre os valores das unidades de necessidade que ultrapassarem um determinado limiar. Na Figura 6.11, é ilustrada a arquitetura geral do sistema motivacional da arquitetura proposta.

O sistema motivacional funciona da seguinte maneira. Inicialmente, os estímulos detectados no ambiente são enviados ao módulo de controle de conseqüências. Então, o pré-processador (*Pré-processador*) codifica estes estímulos para construir um padrão de entrada apropriado ao sistema. Este padrão de entrada pode ou não ser normalizado, dependendo do intervalo numérico dos pesos das conexões selecionadas e do domínio da aplicação da arquitetura. Depois, as unidades de necessidade calculam as respectivas ativações empregando a Equação (6.9) e as respectivas saídas empregando a Equação

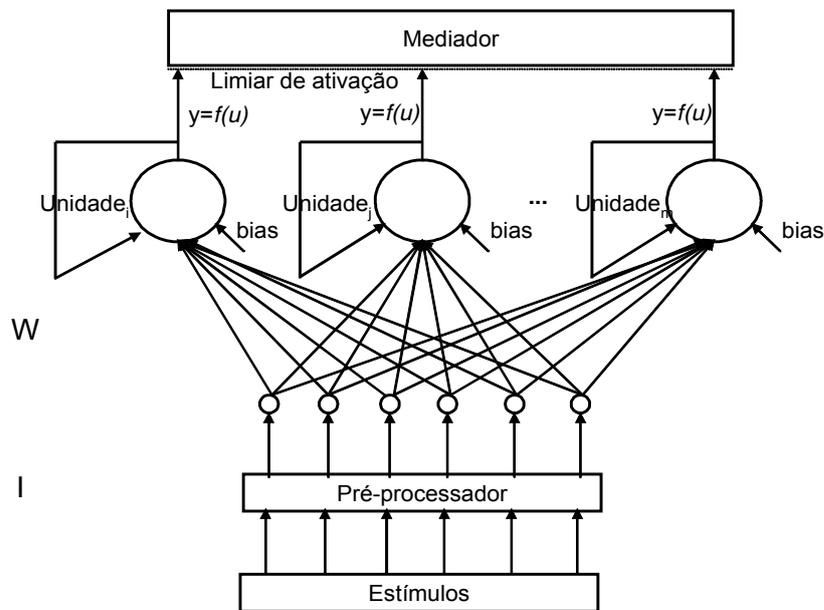


Figura 6.11: Arquitetura do sistema motivacional. O sistema é modelado como uma rede neural artificial competitiva com conexões recorrentes.

(6.10). Posteriormente, o mediador (*Mediador*) seleciona o valor de saída do sistema motivacional, empregando um esquema competitivo e um limiar de ativação. Este mecanismo seleciona a unidade que apresentar maior saída, acima do limiar de ativação. Depois disto, o sistema motivacional checa e informa se qualquer unidade de necessidade recebeu um reforço, por meio da redução de seu valor de ativação.

Aprendizagem de Contingências

A arquitetura proposta pode simular a aprendizagem de contingências e discriminações de estímulos por meio de histórias de reforçamento. A aprendizagem é efetuada por meio de um algoritmo de aprendizado por reforço não determinístico (Sutton and Barto, 1998) (Mitchell, 1997b), pela construção e armazenagem de novas regras de comportamento, assim como pela atualização da probabilidade de execução das regras existentes. O algoritmo de controle e aprendizagem é apresentado pelo Algoritmo 3.

Durante uma interação, o módulo de percepção de estímulos adquire e codifica o estado do ambiente e disponibiliza esta codificação para os módulos de emissão de respostas e o controle de conseqüências. Então, o módulo de controle de conseqüências atualiza e verifica o estado interno do robô e aponta as necessidades ativas, se existirem. Posteriormente, o controle da arquitetura entra em um ciclo de iterações que pode ser executado até o robô alcançar sua metas, ou enquanto durar um episódio de aprendizado. Em cada

Algoritmo 3 Aprendizagem por Contingência.

Requer: um conjunto de estímulos I ($I = \{i_1, i_2, \dots, i_n\}$) que são relacionados para formar o conjunto de estados S ($S = \{s_1, s_2, \dots, s_n\}$);

Um conjunto de ações ($A = \{a_1, a_2, \dots, a_n\}$);

Um conjunto de regras de comportamento ($R = \{r_1, r_2, \dots, r_n\}$);

Um conjunto de necessidades ($N = \{n_1, n_2, \dots, n_n\}$) configuradas no sistema motivacional;

Um valor padrão (Q_d) para inicializar todas as aptidões das ações pertencentes ao conjunto A .

função Aprendizagem de Contingências

Inicie todo $a_i \in A$ com Q_d

Obtenha o estado s_t

Obtenha o retorno do sistema motivacional r_t

Obtenha a necessidade ativa do sistema motivacional n_t

para cada passo em uma tentativa **faça**

 Selecione uma ação a_{t+1} segundo a política π

 Execute a ação a_{t+1} segundo a política π

 Obtenha o estado s_{t+1}

 Obtenha o retorno do sistema motivacional r_{t+1}

 Obtenha a necessidade ativa do sistema motivacional n_{t+1}

se $a_{t+1} \text{ not } \in R$ **então**

 Inicialize o valor da aptidão $Q_i(t+1) = Q_d$

senão

 Atualize o valor da aptidão $Q_i(t+1) = \alpha_n \times (P \times \frac{C_r}{C_n}) + (1 - \alpha_n) \times Q_i(t)$

fim se

fim para

ciclo, o módulo de emissão de respostas utiliza o estado e a informação das necessidades ativas para selecionar uma resposta a ser emitida pelo robô. A seleção de respostas é efetuada de uma maneira estocástica, baseada em uma roda de roleta (Goldberg, 1989). O algoritmo de seleção da roleta possibilita um *bias* nulo e a probabilidade de seleção de uma regra é proporcional ao valor de aptidão da mesma. A distribuição das probabilidades de seleção das regras de comportamento e respostas inatas, proposta nesta tese, é dada por:

$$p_i = \frac{Q_i \pm I}{\sum_{j=1}^n f_j} \quad (6.11)$$

na qual Q_i e Q_j são valores de aptidão de cada resposta ou regra de comportamento. Para tanto, todas as respostas do repertório do robô possuem um valor padrão de aptidão (Q_d) e este valor é pré-definido como um parâmetro na arquitetura. Este valor padrão de aptidão, assim como os valores de aptidão das regras de comportamento, são empregados para construir a roleta de seleção. Adicionalmente, durante o processo de montagem da roleta de seleção, o método de seleção de resposta pode aumentar ou diminuir temporariamente o valor de aptidão de uma regra de comportamento, por meio de uma taxa de influência, quando uma regra satisfaz uma necessidade ativa, ou quando uma regra satisfaz uma necessidade inativa. Esta taxa de influência é determinada pelo sistema motivacional. Ela reflete o estado interno do robô e é dada pela diferença entre o valor de ativação de uma unidade de necessidade e o limiar de ativação das necessidades do sistema motivacional. Desta maneira, a influência é positiva se a unidade de necessidade apontar um estado de privação, e negativa se a unidade de necessidade apontar um estado de saciação.

Após a seleção da resposta mais apropriada, esta é emitida pela execução de uma rotina de motor. Então, o módulo de percepção de estímulos adquire e codifica o novo estado do ambiente e o envia ao módulo de controle de conseqüências e ao módulo de emissão de respostas. O módulo de controle de conseqüências propaga o novo estado codificado pelo sistema motivacional e verifica o estado interno do robô e qualquer reforço recebido como conseqüência da última resposta emitida.

Se a última resposta emitida não for uma regra, o algoritmo de aprendizado então recupera os três termos da contingência (estímulo antecedente, última resposta emitida e conseqüência), da memória de trabalho, e armazena este conhecimento na forma de uma nova regra de comportamento. Se a última resposta for uma regra de comportamento já existente, a arquitetura atualiza a aptidão utilizando a conseqüência percebida de sua execução. A atualização das aptidões das regras de comportamento existentes ou inicialização da

aptidão de uma nova regra de comportamento é dada pela seguinte regra de aprendizagem, proposta nesta tese:

$$Q_i(t+1) = \alpha_n \times \left(P \times \frac{C_r}{C_n}\right) + (1 - \alpha_n) \times Q_i(t) \quad (6.12)$$

na qual $Q_i(t+1)$ é o novo valor de aptidão no tempo presente, P é a potência de um estímulo reforçador, C_r e C_n são respectivamente o contador de reforços recebidos e o contador de execução da regra de comportamento e α_n é uma taxa de aprendizagem decrescente dada por:

$$\alpha_n = \begin{cases} \lambda & \text{if } C_n \leq N_{Interactions} \\ \frac{\lambda}{(C_n - N_{Interactions})} & \text{if } C_n > N_{Interactions} \end{cases} \quad (6.13)$$

na qual $N_{Interactions}$ denota o número mínimo de execuções de uma regra de comportamento antes de começar a decrementar α e λ é uma constante de aprendizagem, ambos parâmetros da arquitetura. Esta taxa de aprendizagem decrescente permite a convergência do algoritmo a uma política ótima. A constante de aprendizagem pode tomar os valores: $0 \leq \lambda < 1$. Se $\lambda = 1$, obtém-se um algoritmo determinístico de aprendizagem.

Esta regra de aprendizagem permite o aumento de um valor de aptidão quando uma regra de comportamento recebe um reforço, assim como permite a redução de um valor de aptidão quando uma regra de comportamento não recebe um reforço que satisfaça as necessidades internas simuladas pelo sistema motivacional. O valor de Aptidão f_t pode variar em um intervalo $[-\infty, +\infty]$.

Na Figura 6.12 é ilustrado o processo de aprendizagem de uma contingência. Inicialmente, o robô está olhando para algum objeto localizado no ambiente. Então, quando o estado interno do robô aponta a necessidade de socialização, o robô procura por um ser humano até detectar um. Se o ser humano estiver mantendo o seu foco de atenção no robô, estabelecendo o contato ocular com este, o robô sente a satisfação de sua necessidade de socialização (por meio do estímulo reforçador da atenção do humano configurado na base de conhecimento da arquitetura). Desta forma, a arquitetura detecta o reforço recebido como consequência da emissão de resposta (ilustrado como “0 get(attention)” na Figura 6.12). Então, a arquitetura recupera a última resposta emitida e todos os estímulos antecedentes para criar uma nova regra de comportamento que representa esta contingência. Para tanto, a arquitetura atribui um valor igual a 1 aos contadores de reforço e de execução e calcula o valor da aptidão da nova regra, armazenando também a necessidade satisfeita pela execução de mesma.

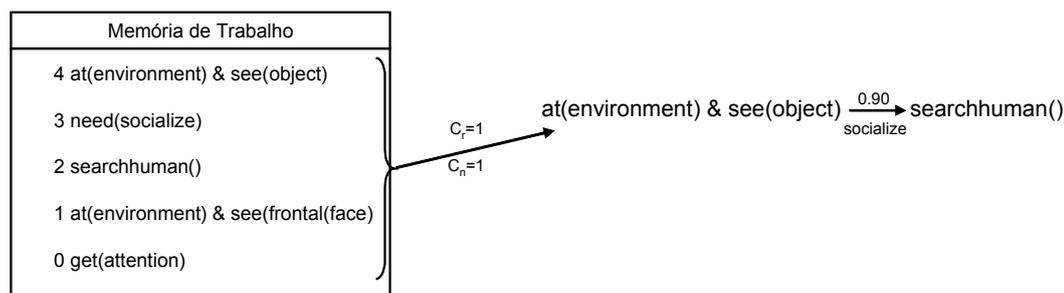


Figura 6.12: Exemplo do processo de aprendizagem de contingências. A chave indica todos os itens da memória de trabalho utilizados para criar a nova regra de comportamento. C_r e C_n são os contadores associados a nova regra, inicialmente iguais a 1. Para este exemplo, P é igual a 1 e λ é igual a 0.9. A seta representa o processo de criação da nova regra.

Modelagem e Encadeamento de Comportamentos

A arquitetura proposta permite a modelagem do comportamento do robô por meio de aproximações sucessivas, em um processo de reforçamento dos comportamentos inatos do robô, constituindo um processo de discriminação operante. Por exemplo, se for requerido que o robô aprenda a seguir o olhar de um ser humano, pode-se modelar inicialmente o seu comportamento de manter contato ocular com o ser humano. Deste modo, o robô aprende que pode socializar procurando seres humanos e mantendo o contato ocular com estes. Este processo pode ser repetido de forma que novas regras de comportamento sejam criadas e a probabilidade de emitir determinadas respostas em determinados contextos seja aumentada ou reduzida gradualmente.

Uma vez que o robô tenha aprendido a manter o contato ocular com o ser humano, este pode aprender a seguir o seu olhar para algum local no ambiente, para o qual o ser humano esteja olhando. Assim, quando o robô mantiver contato ocular com o ser humano e este desviar sua atenção (mover sua cabeça) para outro objeto, o robô perderá o estímulo reforçador da atenção do ser humano e iniciará uma busca, procurando algum estímulo interessante no ambiente. Quando o robô localizar e focar algum objeto que seja o foco de atenção do ser humano, este poderá voltar a receber a atenção deste humano, por meio de verbalizações sobre o objeto de interesse em comum, constituindo um estímulo reforçador da atenção do ser humano para com o robô (configurado na base de conhecimento da arquitetura). Neste momento, a arquitetura aprende a seguir o olhar do ser humano, como ilustrado na Figura 6.13.

Desta maneira, a arquitetura pode simular o processo de aprendizagem de contingências por um indivíduo durante uma interação com o ambiente.

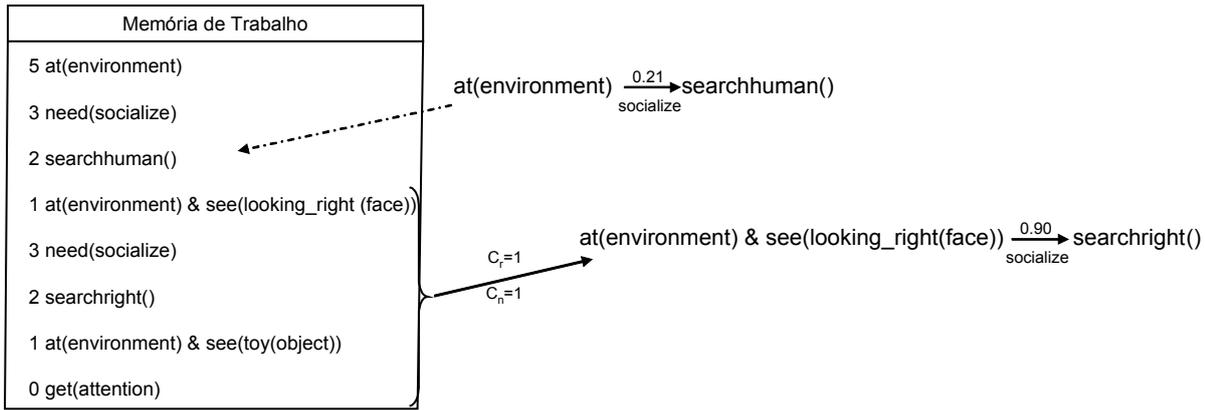


Figura 6.13: Exemplo de encadeamento de comportamentos. A chave indica todos os predicados empregados para se criar uma nova regra de comportamento. Os valores C_r e C_n são, respectivamente, o contador de reforço e o contador de execução, associados à regra de comportamento, com valor inicial igual a 1 (hum). A seta pontilhada representa a emissão de resposta pela execução de uma regra previamente aprendida. A seta sólida representa o processo de criação da nova regra de comportamento.

6.6.6 Convergência para uma Política Ótima

Nesta seção é efetuada uma discussão sobre a capacidade de convergência do algoritmo de aprendizagem da arquitetura básica para uma política ótima. Para isso, deve-se recorrer ao trabalho apresentado por Mitchell (Mitchell, 1997b), no qual o autor demonstra que um agente de aprendizagem em um processo de decisão de Markov (MDP) não determinístico pode alcançar a convergência para uma política ótima, desde que atenda às condições do seguinte Teorema:

Teorema 1. (Mitchell, 1997b)

Considere um agente de aprendizagem em um MDP não determinístico, com recompensas restritas por $(\forall s, a) |r(s, a)| \leq c$. No qual o agente de aprendizagem utiliza uma regra de aprendizagem dada pela Equação (6.14), inicializa sua tabela Q com valores finitos arbitrários, e utiliza um fator de desconto γ tal que $0 \leq \gamma \leq 1$. Seja $n(i, s, a)$ a interação que corresponde ao i th tempo que a ação a é aplicada ao estado s . Se cada par é visitado freqüentemente e infinitamente, $0 \leq \alpha < 1$, e

$$\sum_{i=1}^{\infty} \alpha_{n(i,s,a)} = \infty, \quad \sum_{i=1}^{\infty} [\alpha_{n(i,s,a)}]^2 < \infty$$

Então, para todo s e a , o algoritmo converge para uma política ótima, quando $n \rightarrow \infty$, com probabilidade igual a 1.

$$Q_n(s, a) \leftarrow (1 - \alpha_n) \times Q_{n-1}(s, a) + \alpha_n \times [r + \gamma \max_{a'} Q_{n-1}(s', a')] \quad (6.14)$$

Baseado no Teorema 1, pode-se então argumentar que o algoritmo de aprendizado da arquitetura proposta pode convergir para uma ótima política, como declarado no Teorema seguinte:

Teorema 2. *O processo de aprendizagem dado pelas Equações (6.12) e (6.13) converge para uma ótima política com probabilidade 1.*

Prova

Deste que α_n é definido pela Equação (6.13), então:

$$\lim_{n \rightarrow \infty} \alpha_n = 0$$

Desde que f_n é dado pela Equação (6.12), então:

$$\begin{aligned} \lim_{n \rightarrow \infty} f_n &= 0 \times (P \times \frac{C_r}{C_n}) + (1) \times f_{n-1} \\ &= S_{n-1} \end{aligned}$$

na qual

$$S_{n-1} = \sum_{j=1}^{n-1} f_j$$

Adicionalmente, pela redução de α_n a uma taxa apropriada, durante um episódio de treinamento, o algoritmo pode alcançar a convergência a uma política ótima, desde que respeitadas as condições do Teorema 1. Neste contexto, a seguir é então demonstrado que o algoritmo de aprendizagem da arquitetura atende às condições do Teorema.

Em primeiro lugar, os reforços são restritos $(\forall s, a) |r(s, a)| \leq 1$ desde que λ esteja configurado com um valor menor que 1 na arquitetura. Além disso, a regra de aprendizagem da arquitetura proposta pode ser obtida fazendo $\gamma = 0$ na Equação (6.14). Adicionalmente, o algoritmo de aprendizagem inicializa seus valores de reforço com valores finitos arbitrários (Q_d). Finalmente, escolhendo $\lambda < 1$ e utilizando-se a teoria de séries harmônicas, pode-se verificar a satisfação da seguinte condição:

$$\sum_{i=1}^{\infty} \alpha_{n(i,s,a)} = \infty, \quad \sum_{i=1}^{\infty} [\alpha_{n(i,s,a)}]^2 < \infty$$

Portanto, o algoritmo de aprendizado da arquitetura converge para uma política ótima com probabilidade igual a 1.

A seguir são apresentados os experimentos realizados durante esta pesquisa.

6.7 Simulação da Caixa de Skinner

Nesta seção, são apresentados os principais resultados dos experimentos executados para a avaliação preliminar e prova de conceito da primeira versão da arquitetura proposta. O propósito destes experimentos foi demonstrar o funcionamento das estruturas e mecanismos da arquitetura, bem como avaliar preliminarmente a sua capacidade de aprendizagem. Estes experimentos foram executados empregando-se um simulador simples de uma Caixa de Skinner. Este simulador foi desenvolvido como uma máquina de estados na qual cada estado representa uma posição do rato na caixa de Skinner. A caixa é composta por nove quadrantes que representam seus lados e cantos. Em um determinado lado da caixa, existe uma barra, que pode ser utilizada para disponibilizar alimentos. Acima da barra, existe uma luz de controle que pode ser programada para ficar acesa, apagada, ou acender e apagar em intervalos fixos. Durante a simulação, o rato pode explorar qualquer um dos nove quadrantes da caixa e, quando o rato está no lado direito da caixa, este pode ver a barra, a luz de controle e o comedouro de alimentos. Na Figura 6.14 pode ser visto um esquema do simulador.

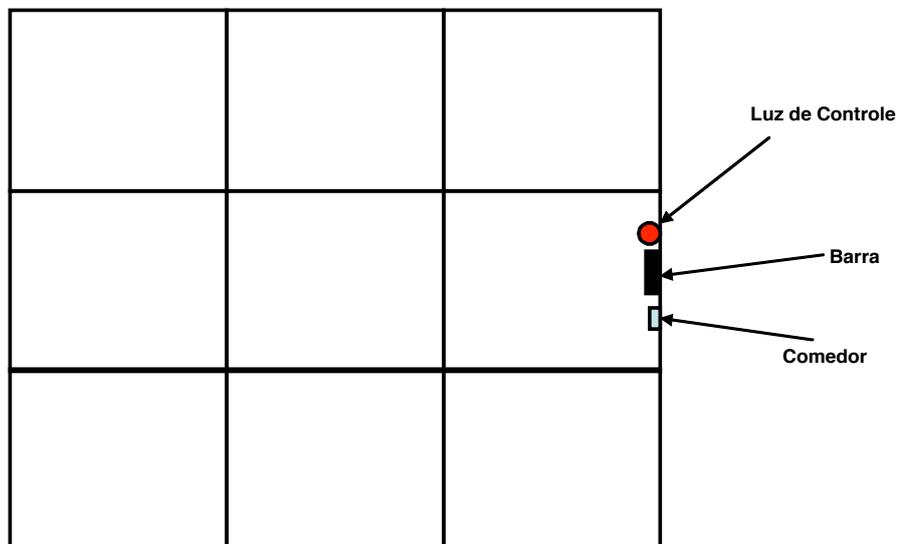


Figura 6.14: Representação esquemática do simulador da caixa de Skinner. Os quadrantes representam as posições nas quais o rato pode explorar a caixa de Skinner. O canto direito da caixa possui uma luz de controle, uma barra e um comedouro no qual pode ser disponibilizada uma pelota de alimento.

Neste domínio de problema, o objetivo foi ensinar o rato simulado a pressi-

onar a barra que aciona o comedouro, para obter alimento, somente quando a luz de controle estivesse acesa. O processo de aprendizado foi executado com o controle dos estímulos presentes no ambiente e os reforços das ações corretas do rato, em um processo de aproximações sucessivas. Neste experimento, o simulador foi programado com uma barra de comida. O simulador foi programado para executar a simulação por quatro mil unidades de tempo. A luz de controle foi programada para ficar acesa nas primeiras quinhentas unidades de tempo e então começar a alternar entre acesa e apagada até o final da simulação.

Nesta primeira versão da arquitetura, a regra de aprendizagem era ligeiramente diferente da versão atual da regra apresentada na Seção 6.6 (ver Equação 6.12) e esta possuía alguns parâmetros adicionais que foram eliminados nas versões posteriores e portanto não são descritos na apresentação da arquitetura. Estes parâmetros adicionais eram empregados para limitar os valores de aptidão que uma regra poderia assumir, ou seja, os valores de aptidão de uma regra (f_i) podiam variar no intervalo $[\tau_i, \tau_s]$, no qual τ_i e τ_s eram parâmetros empiricamente determinados. Entretanto, após o estudo de convergência do algoritmo de aprendizado e a modificação da regra de aprendizado, o valor da aptidão passou a variar livremente no intervalo $[-\infty, +\infty]$ (como apresentado em 6.6) e os parâmetros τ_i e τ_s foram eliminados.

Adicionalmente, esta primeira versão da arquitetura era composta por outros dois parâmetros que definiam a taxa de aumento ou redução temporária do valor de aptidão de uma regra. Estes parâmetros eram definidos como σ_+ e σ_- e eram empregados durante o processo de montagem da roleta de seleção para alterar temporariamente o valor de aptidão de uma regra da seguinte maneira. Se uma regra satisfizesse uma necessidade ativa, o mecanismo de seleção de respostas incrementava o valor da aptidão da regra, adicionando ao mesmo o valor definido em σ_+ . Se uma regra satisfizesse uma necessidade inativa, o mecanismo de seleção de respostas decrementava o valor da aptidão da regra, subtraindo do mesmo o valor definido em σ_- . Após alguns estudos adicionais, estes parâmetros foram eliminados da arquitetura e substituídos pela taxa de influência que é determinada pelo sistema motivacional. Portanto estes não são apresentados na Seção 6.6 (ver Equação 6.11).

Para este experimento, a base de conhecimento de arquitetura foi configurada como a seguir. Foram declarados treze estímulos a serem detectados no ambiente: *ne_corner*, *se_corner*, *nw_corner*, *sw_corner*, *nt_side*, *st_side*, *et_side*, *wt_side*, *center*, *light_on*, *light_off*, *bar* e *food*, entre os quais, *food* é um estímulo reforçador.

A arquitetura foi configurada da seguinte maneira. A constante de aprendizagem (parâmetro λ) foi configurada com valor igual a 0.5. O número mínimo

de execuções antes de iniciar o decréscimo da taxa de aprendizagem (parâmetro $N_{Interactions}$) foi configurado com valor igual a 1. O valor padrão de aptidão (parâmetro Q_d) foi configurado com valor igual a 0.001. Os limites dos valores máximo e mínimo de aptidão que a arquitetura (parâmetros τ_i e τ_s) foram configurados, respectivamente, com valores iguais a 0.03 e 0.8. As taxas de acréscimo e decréscimo temporário da aptidão (parâmetros σ_+ e σ_-) foram configurados, respectivamente, com valores iguais a 0.4 e 0.8.

O sistema motivacional foi configurado com uma unidade de necessidade que simula a fome (*hunger*) e seu padrão de entrada do sistema foi configurado para reconhecer o estímulo *food* como reforçador. O limiar de ativação do sistema motivacional foi configurado com um valor igual a 0.7. A inclinação da função sigmóide (parâmetro δ) foi configurada com um valor igual a 0.20. O *bias* da unidade de ativação foi configurado com um valor igual a 1.50 e o peso de sua conexão foi configurado com um valor igual a 0.15. O peso da conexão recorrente foi configurado com um valor igual a 1.00. Os pesos das conexões das unidades de entrada (*see(food)* e *smell(food)*) foram configurados, respectivamente, com valores iguais a -1.50 e 0.10 . Durante o processo de configuração da arquitetura, verificou-se empiricamente que estes valores produziram os melhores resultados.

Durante este experimento, o rato foi colocado em um lugar arbitrário da caixa e começou a explorar a mesma. Inicialmente, quando o rato explorava o lado da caixa próximo a barra do comedouro, uma pelota de comida era produzida pelo simulador. Imediatamente, a arquitetura detectava a pelota e produzia um reforçamento pela satisfação, mesmo que parcial, da unidade de necessidade que simulava a fome (*hunger*). Este procedimento foi repetido durante cinco vezes para ensinar ao rato a explorar o lado próximo a barra para receber alimento. Desta forma, sempre que o rato tinha fome, este ia para perto da barra. Em uma destas explorações, o rato pressionou incidentalmente a barra do comedouro e obteve uma pelota de alimento (com a luz de controle acesa). Este processo foi repetido algumas vezes e assim o rato aprendeu a pressionar a barra para obter uma pelota de alimento.

Posteriormente, quando a luz de controle passou a alternar entre acesa e apagada, o rato continuou a pressionar a barra do comedouro mas, quando a luz de controle estava apagada, este não recebia alimento. Neste processo, a arquitetura aprendeu que quando a luz de controle está apagada a pressão da barra do comedouro não produz as pelotas de alimento. Desta forma, o rato simulado aprendeu a pressionar a barra somente quando a luz de controle está acesa. Na Figura 6.15, é apresentado o comportamento do rato durante a simulação. O gráfico mostra a influência do sistema motivacional no comportamento do rato. Quando a saída do sistema motivacional ultra-

passa o limiar de ativação (*Activation Threshold*), o rato entra em um estado de privação de alimento e aumenta a probabilidade de emitir um comportamento que satisfaça esta necessidade. O gráfico demonstra que o rato emitiu o comportamento de pressão à barra, em geral, sempre que a saída do sistema motivacional ultrapassou o limiar de ativação. O gráfico demonstra também que o rato emitiu o comportamento de pressão à barra, na maioria das vezes, quando a luz de controle estava acesa.

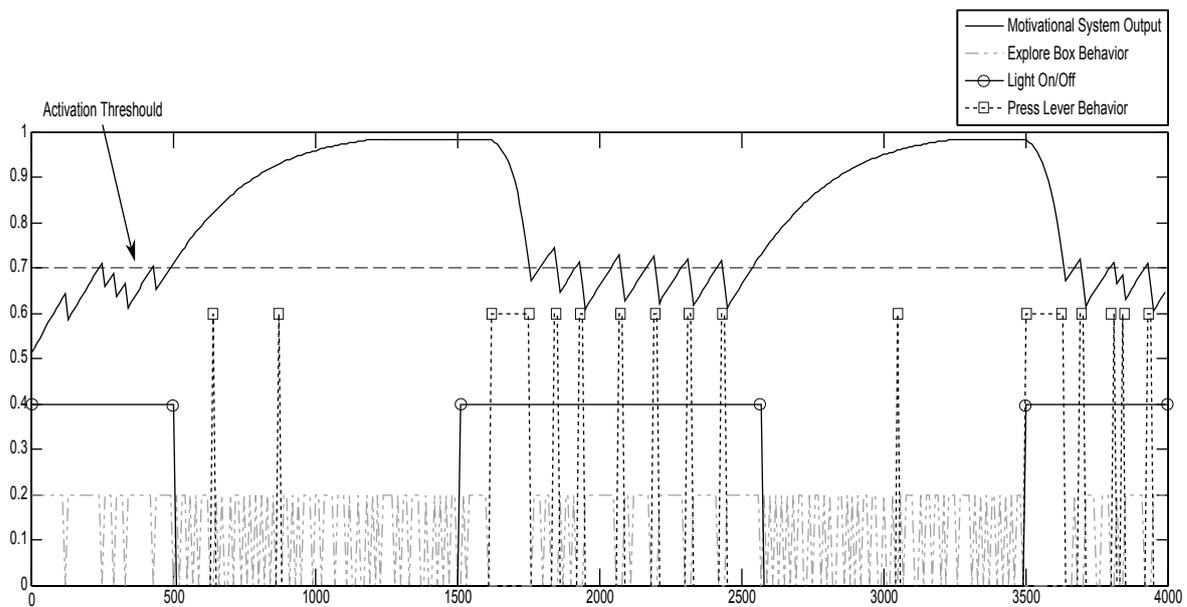


Figura 6.15: Influência do sistema motivacional no comportamento do rato. O *Activation Threshold* é o limiar empregado pelo sistema motivacional para verificar se alguma necessidade dominante está ativa. Todas as saídas foram normalizadas para serem ajustadas à escala do gráfico.

Embora este experimento preliminar seja simples e possua um espaço de busca e decisão reduzido, estes resultados preliminares demonstram que a arquitetura pode controlar o comportamento de um agente sociável e emitir comportamentos apropriados em uma interação com o ambiente. Os resultados também mostram que a arquitetura suporta a modelagem de comportamento e é capaz de estabelecer um processo de encadeamento, isso é, pode-se modelar o comportamento de um agente por meio de aproximações sucessivas do comportamento desejado, pelo reforço de comportamentos inatos para a formação de uma cadeia de comportamentos. Neste experimento, isto foi realizado pelo reforçamento da exploração do lado da caixa próximo à barra do comedouro e depois pelo reforçamento da própria pressão da barra, criando uma cadeia de comportamentos na qual o rato aprendeu a explorar o lado próximo à barra e então pressioná-la, sempre que estiver em privação de alimento. Estes processos de modelagem do comportamento e ensino são

importantes processos evidenciados na Análise do Comportamento.

Finalmente, um exame dos resultados apresentados demonstra que o sistema motivacional foi capaz de influenciar positivamente no processo de emissão das respostas pelo rato simulado. Estes resultados estão de acordo com os apresentados na literatura específica sobre sistemas motivacionais para robô sociáveis (Breazeal, 2000) (Breazeal and Scassellati, 2000) (Breazeal, 2002) (Breazeal, 2003a).

Como resultado deste experimento, setenta e nove regras de comportamento foram geradas. Foram criadas diversas regras de comportamento para controlar a exploração do rato na caixa:

...

$$\text{at}(\text{ne_side}) \xrightarrow{0.003} \text{explore}(\text{se_side})$$
 ...

Algumas regras de comportamento foram criadas para controlar a exploração do lado perto da barra, quando estava em estado de privação e alimento:

...

$$\text{at}(\text{ne_side}) \xrightarrow[\text{hunger}]{0.0699} \text{explore}(\text{et_side})$$

$$\text{at}(\text{st_side}) \xrightarrow[\text{hunger}]{0.0572} \text{explore}(\text{et_side})$$

$$\text{at}(\text{nw_corner}) \xrightarrow[\text{hunger}]{0.2099} \text{explore}(\text{et_side})$$
 ...

Adicionalmente, foram criadas duas regras de comportamento para controlar o comportamento de pressionar a barra quando o rato estava em estado de privação de alimento:

$$\text{at}(\text{et_side}) \& \text{see}(\text{light_on}) \& \text{see}(\text{bar}) \xrightarrow[\text{hunger}]{0.799} \text{press}(\text{bar})$$

$$\text{at}(\text{et_side}) \& \text{see}(\text{light_off}) \& \text{see}(\text{bar}) \xrightarrow{0.0075} \text{press}(\text{bar})$$

A primeira regra de comportamento modela o comportamento do rato para pressionar a barra do comedouro para obter uma pelota de alimento, com uma alta probabilidade, sempre que a luz de controle está acesa. A segunda regra de comportamento modela o comportamento do rato para pressionar a barra do comedouro para obter uma pelota de alimento, com uma baixa probabilidade, sempre que a luz e controle está apagada. Desta forma, pode-se concluir que o rato aprendeu a pressionar a barra do comedouro somente quando a luz de controle estivesse acesa.

Após estes primeiros resultados obtidos com os experimentos preliminares

empregando a arquitetura e o simulador da caixa de Skinner, foi realizado um estudo sobre a convergência do algoritmo de aprendizagem, além de um estudo para a modificação de algumas estruturas que possibilitassem a eliminação de alguns parâmetros existentes, tornando a arquitetura mais auto-adaptável e mais simples de configurar. Como resultado dos estudos sobre a convergência, foi efetuada uma alteração na regra de aprendizagem utilizada pela arquitetura, resultando na versão atual apresentada na Seção 6.6 (ver Equação 6.12). Adicionalmente, foram eliminados os parâmetros τ_i e τ_s que limitavam os valores de aptidão das regras de comportamento, uma vez que estes valores passaram a convergir dentro de intervalos controlados pela própria convergência do algoritmo, em intervalos controlados pela constante de aprendizagem (parâmetro λ). Ainda, como resultado sobre os estudos para a eliminação sobre as estruturas da arquitetura, foram eliminados os parâmetros σ_+ e σ_- , que eram empregados durante o processo de montagem da roleta de seleção para alterar temporariamente o valor de aptidão de uma regra da seguinte maneira. Após estes estudos, estes parâmetros foram substituídos pela taxa de influência (I) que é controlada pelo sistema motivacional, como apresentado na Seção 6.6.

Posteriormente a estas modificações na arquitetura, foram realizados novos experimentos para validar a nova versão da mesma, empregando-se o simulador da caixa de Skinner. Entretanto, o objetivo deste novo experimento foi validar a capacidade da arquitetura em aprender a executar a tarefa proposta (pressionar a barra somente com a luz de controle acesa).

Para este novo experimento, a base de conhecimento da arquitetura foi configurada como no experimento anterior (ver página 6.7).

Para este novo experimento foi executado o mesmo processo de simulação descrito no experimento anterior. O rato foi colocado em um lugar arbitrário da caixa e começou a explorar a mesma. Quando o rato explorava o lado da caixa próximo a barra do comedouro, uma pelota de comida era produzida. Imediatamente, a arquitetura detectava a pelota e produzia um reforçamento pela satisfação, mesmo que parcial, de uma unidade de necessidade (*hunger*). Este procedimento foi repetido durante cinco vezes para ensinar ao rato a explorar o lado próximo a barra para receber alimento. Desta forma, sempre que o rato tinha fome, este ia para perto da barra. Em uma destas explorações, o rato pressionou a barra do comedouro e obteve uma pelota de alimento (com a luz de controle acesa). Este processo foi repetido algumas vezes e assim o rato aprendeu a pressionar a barra para obter uma pelota de alimento. Posteriormente, quando a luz de controle passou a alternar entre acesa e apagada, o rato continuou a pressionar a barra do comedouro. Entretanto, quando a luz de controle estava apagada, este não recebia alimento. Neste

processo, a arquitetura aprendeu que quando a luz de controle está apagada a pressão da barra do comedouro não produz as pelotas de alimento. Desta forma o rato simulado aprendeu pressionar a barra somente quando a luz de controle está acesa, confirmando os resultados obtidos pelo experimento anterior.

A metodologia seguida para este novo experimento foi diferente da primeira, uma vez que se objetivou levantar a curva de aprendizado do rato para validar a capacidade de aprendizagem da arquitetura. Esta metodologia é descrita na Seção 6.4. Este experimento foi composto por uma fase de aprendizagem com duração de 4.000 unidades de tempo, ou 4.000 ciclos de simulação.

A capacidade de aprendizado da arquitetura foi analisada pela observação da interação do rato com o ambiente e pela computação de uma métrica denominada *press bar index* ou *índice pressão a barra* (PBI). Esta métrica é definida como a frequência de ações de pressionar a barra de alimentação da caixa de Skinner com a luz de controle acesa. Esta métrica é dada pela Equação (6.2), apresentada na Seção 6.4.

Para quantificar o aprendizado da arquitetura durante os experimentos, a fase de aprendizagem era interrompida em pontos específicos (a cada 500 unidades de tempo) e uma fase de validação do conhecimento adquirido era iniciada para avaliar o comportamento da arquitetura. Esta avaliação era executada por 20 corridas de 500 unidades de tempo (500 ciclos de interação). Para cada corrida, o valor do PBI era computado e, depois das 20 corridas, a média e desvio padrão das 20 medidas eram calculados, dados respectivamente pelas Equações (6.3) e (6.4), apresentadas na Seção 6.4. Depois das 20 corridas da fase de avaliação, a fase de aprendizagem era retomada do ponto no qual esta havia sido interrompida.

Na Figura 6.16, é apresentada a curva de aprendizado que mostra o progresso da aprendizagem durante a simulação. O gráfico apresenta a média da métrica PBI para cada fase de avaliação, em pontos específicos durante a fase de aprendizagem. Este gráfico também mostra a influência da constante de aprendizagem (parâmetro λ) no desempenho da arquitetura. Finalmente, este gráfico mostra a capacidade de aprendizagem da arquitetura, apresentando as curvas com os valores crescentes de PBI a cada fase de validação. Demonstra-se, portanto, que a arquitetura foi capaz de adquirir o conhecimento necessário para executar a tarefa de pressionar a barra de alimento somente com a luz de controle acesa.

A Tabela 6.1 mostra os melhores resultados obtidos empregando as diferentes constantes de aprendizagem.

Os resultados mostram que a constante de aprendizagem não influenciou significativamente o desempenho da arquitetura. Para confirmar estes resul-

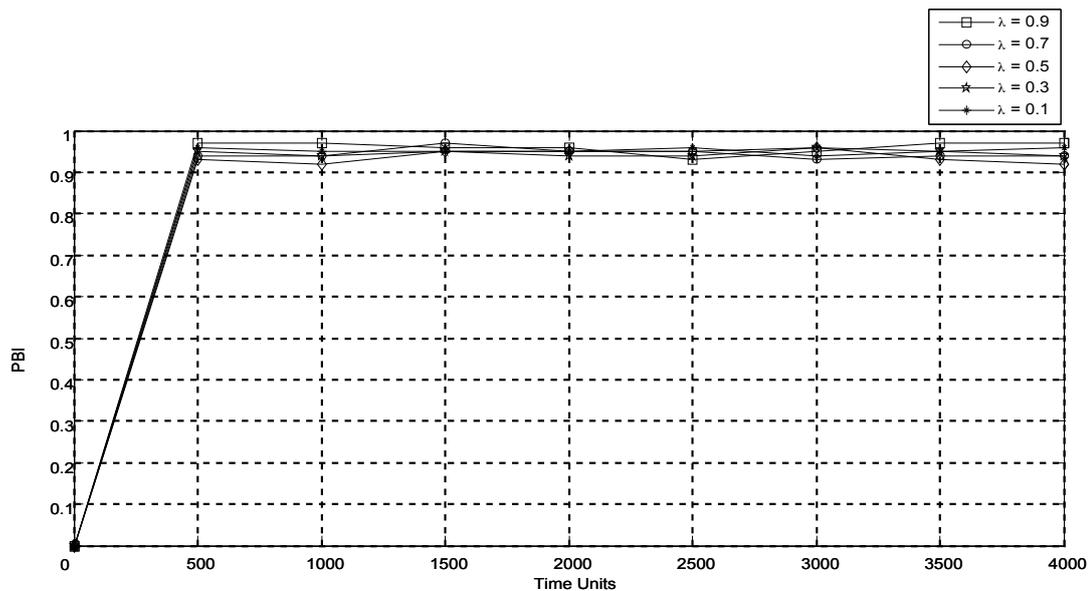


Figura 6.16: Evolução da aprendizagem durante os experimentos. As linhas apresentam as avaliações para diferentes valores da constante de aprendizagem (parâmetro λ). Em pontos específicos, o processo de aprendizagem foi temporariamente interrompido e uma avaliação do comportamento do rato simulado foi executada por meio de 20 corridas de 500 unidades de tempo. Para cada corrida, foi computado o valor do PBI e depois das 20 corridas foram calculadas as suas médias e desvios padrão.

Tabela 6.1: Melhores resultados obtidos durante o processo de aprendizagem.

λ	Melhor média e desvio padrão do PBI
0.9	0.97 ± 0.02
0.7	0.97 ± 0.02
0.5	0.96 ± 0.02
0.3	0.96 ± 0.02
0.1	0.96 ± 0.03

tados, foi utilizado o *teste t* pareado para procedimentos bilaterais com 99% de certeza (Moses, 1986) (Mitchell, 1997b). Os resultados são mostrados na Tabela 6.2.

Tabela 6.2: Resultados do Teste *t* as melhores médias do PBI.

Modelos Comparados	Conclusão
Arquitetura com $\lambda = 0.9$ e Arquitetura com $\lambda = 0.7$	Desempenho Similar
Arquitetura com $\lambda = 0.9$ e Arquitetura com $\lambda = 0.5$	Desempenho Similar
Arquitetura com $\lambda = 0.9$ e Arquitetura com $\lambda = 0.3$	Desempenho Similar
Arquitetura com $\lambda = 0.9$ e Arquitetura com $\lambda = 0.1$	Desempenho Similar

Os resultados obtidos mostram que, de uma maneira geral, a arquitetura

apresentou uma boa capacidade de aprendizado e de exibir comportamentos apropriados durante a simulação. Os resultados também confirmam as conclusões apresentadas sobre o experimento anterior e mostram que a arquitetura é uma ferramenta promissora para o controle de robôs sociáveis em interações em um ambiente social. Finalmente, um exame dos resultados obtidos mostram que o algoritmo de aprendizagem da arquitetura pode constituir uma importante contribuição para a área de aprendizado de robôs sociáveis. Entre suas mais importantes características pode-se citar a sua capacidade de mapear motivações, estímulos discriminativos e conseqüências para produzir uma política de aprendizagem não determinística.

6.8 *Simulação da Atenção Compartilhada*

Após os primeiros experimentos com a arquitetura, empregando o simulador da caixa de Skinner, foram realizadas algumas modificações adicionais na mesma (apresentadas nas próximas Seções) e então foram executados diversos experimentos para a validação da arquitetura em um problema de aplicação real e não trivial, o aprendizado da atenção compartilhada (Dube et al., 2004) (Kaplan and Hafner, 2004). Para tanto, foi desenvolvido um simulador de interações sociais, baseado em trabalhos de Triesch e colegas (Triesch et al., 2006), capaz de simular os movimentos necessários ao robô e ao ser humano, além de fornecer estímulos do ambiente apropriados ao contexto da atenção compartilhada. Este simulador é apresentado a seguir.

6.8.1 *Simulador de Interações Sociais*

Durante o presente trabalho, um simulador de interações sociais foi desenvolvido para os experimentos da atenção compartilhada que é capaz de simular uma interação entre um robô e um ser humano em um ambiente social controlado. Na Figura 6.17, é apresentada a interface do simulador desenvolvido.

Para simular a atenção compartilhada, foram definidas três entidades que podem ser manipuladas por funções do simulador, um humano, um robô e um brinquedo. Neste simulador, o ser humano e o robô foram posicionados frente a frente a uma distância fixa. Além do ser humano e do robô, o simulador possibilita posicionar até dois brinquedos simultaneamente no ambiente social. Esta funcionalidade é útil para se simular um objeto distrator enquanto se posiciona um objeto que é foco de atenção do ser humano, permitindo verificar se o robô olha para o brinquedo correto, mesmo na presença de outros objetos no ambiente. Um brinquedo pode ser posicionado em qualquer lugar vazio do ambiente social, a qualquer momento durante uma simulação.

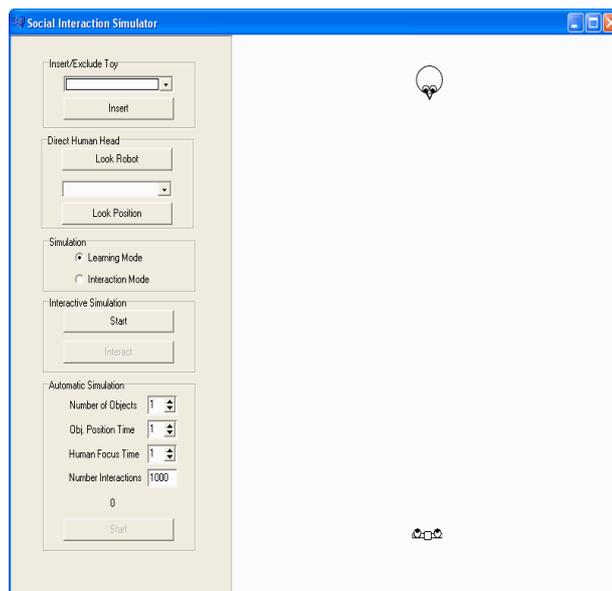


Figura 6.17: Interface do simulador de interações sociais. No lado esquerdo da interface está o painel de controle que habilita as simulações interativas ou automáticas. O ser humano é fixo na porção superior da interface e é capaz de girar a sua cabeça em um ângulo de ± 90 graus. O robô é fixo na porção inferior da interface e também é capaz de girar a sua cabeça em um ângulo de ± 90 graus.

O ambiente social foi modelado da seguinte maneira. Tanto o robô como o ser humano podem girar suas cabeças para a esquerda ou para a direita, em um ângulo de até 90° . O robô tem seu foco central em 0° e tem seu campo visual limitado por um parâmetro de fôvea λ° , permitindo que o robô visualize objetos em um campo visual formado por um cone de abertura dada por: $[-\lambda^\circ, +\lambda^\circ]$, com centro em 0° . Na Figura 6.18, é ilustrada esta modelagem do campo visual do robô.

A posição da cabeça do robô é determinada por θ_r , que pode assumir valores entre $[-90^\circ, +90^\circ]$. A posição da cabeça do ser humano é determinada por θ_a , que também pode assumir valores entre $[-90^\circ, +90^\circ]$. Quando um objeto i é posicionado no ambiente social, o simulador traça o ângulo entre este objeto e o foco do robô, ou seja, ele determina o deslocamento da cabeça do robô necessário para focalizar o objeto posicionado no ambiente. Esta mapeamento é dado por θ_{oi} , que pode assumir valores entre $[-90^\circ, +90^\circ]$. Desta maneira, se um objeto é posicionado no ambiente, o simulador verifica se o mesmo está dentro do campo visual do robô, comparando sua posição em relação ao foco da visão do robô, considerando o campo de visão do mesmo. Na Figura 6.19, são ilustrados os parâmetros de posicionamento dos objetos e do ser humano em relação ao campo visual do robô.

Adicionalmente aos estímulos visuais da face do ser humano e dos brin-

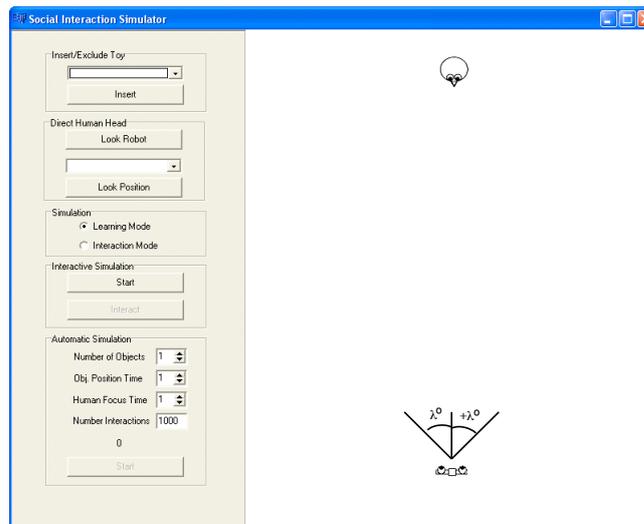


Figura 6.18: Campo visual do robô. As linhas representam os limites do campo visual do robô, com abertura dada por: $[-\lambda^\circ, +\lambda^\circ]$, com centro em 0° .

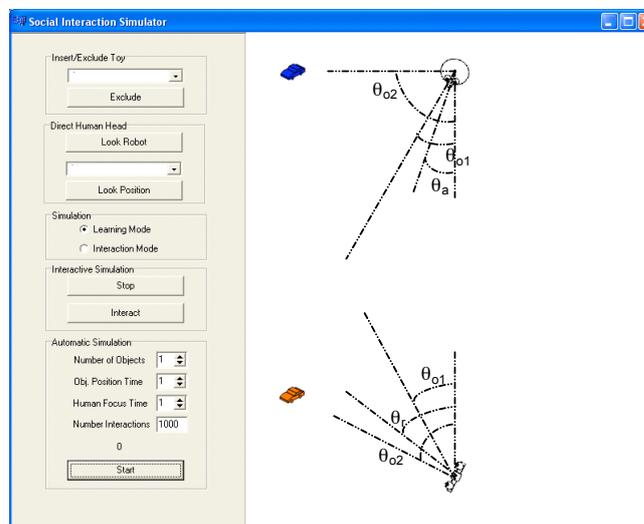


Figura 6.19: Controle de posicionamento. As linhas representam as distâncias entre o foco do robô e os objetos posicionados no ambiente social, assim como a posição da cabeça do robô e da cabeça do ser humano. Nesta figura é mostrada uma interação na qual o ser humano está olhando para um objeto posicionado no ambiente e o robô acompanha seu olhar para o objeto correto, apesar da existência de um objeto distrator.

quedados, o simulador provê um estímulo auditivo que simula a atenção do ser humano para com o robô. O simulador provê este estímulo quando o humano e o robô estão mantendo contato ocular e quando o robô segue o olhar do ser humano até um objeto correto, que é foco de atenção deste último. Este mecanismo foi incorporado no simulador para simular os resultados da análise comportamental apresentada por Dube e seus colegas (Dube et al., 2004), na qual eles argumentam que os adultos agem como operadores motivacionais no contexto da aprendizagem de atenção compartilhada, fornecendo uma classes de estímulos reforçadores denominados de *atenção do ser humano*, como apresentado na Seção 6.3.

Durante uma simulação, o simulador é capaz de executar interações continuamente e cada interação toma aproximadamente 1 segundo. O simulador pode posicionar até dois objetos simultâneos no ambiente social, em posições estocasticamente selecionadas com probabilidade ρ_o . Estes objetos são posicionados nos respectivos lugares durante um tempo determinado pelo usuário (determinado em segundos ou ciclos de interações no painel de controle). Adicionalmente, o simulador pode direcionar a cabeça do ser humano para focar um objeto presente no ambiente ou para focar o robô. O objeto que recebe o foco do olhar do ser humano é estocasticamente selecionado com probabilidade ρ_a . Após focar um objeto, o humano permanece com o seu foco no mesmo durante um tempo determinado pelo usuário (determinado em segundos ou ciclos de interações no painel de controle), antes de direcionar a sua cabeça para focar outro objeto ou para focar o robô.

6.8.2 Principais Resultados

Após os resultados obtidos com os experimentos empregando a arquitetura e o simulador da caixa de Skinner, foi realizada uma pequena modificação no algoritmo de recuperação das regras de comportamento para otimizar o tempo de recuperação das mesmas. Então, um novo conjunto de experimentos foi executado para avaliar a arquitetura, empregando simulador de interações sociais.

Nesta Seção, são apresentados os principais resultados dos experimentos executados para a avaliação da arquitetura proposta no contexto do aprendizado da atenção compartilhada, empregando o simulador de interações sociais apresentado anteriormente. O propósito destes experimentos foi a determinação da capacidade de exibição de comportamentos apropriados e de aprendizagem da arquitetura durante o controle do robô simulado em um ambiente social controlado.

Para este conjunto de experimentos, o conhecimento de arquitetura foi configurado da seguinte maneira. Quatro estímulos foram declarados: *face*,

object, *attention* e *environment*, no qual *attention* é um estímulo reforçador gerado com a atenção do ser humano. Foram declarados dois fatos para definir que objetos vermelhos e azuis são brinquedos. Também foram declarados treze fatos para diferenciar a pose da cabeça do ser humano como frontal, além de seis poses de perfil esquerdo e seis poses de perfil direito. Adicionalmente, foram declarados mais fatos para definir quando o robô está focalizando o ser humano ou um brinquedo.

O módulo de emissão de respostas foi configurado como a seguir. A constante de aprendizagem (parâmetro λ) foi variada para determinar a sua influência no desempenho da arquitetura e recebeu os seguintes valores: $\lambda = \{0.1, 0.3, 0.5, 0.7, 0.9\}$. O número mínimo de execuções das regra de comportamento antes de se iniciar o decréscimo da taxa de aprendizagem (parâmetro $N_{Interactions}$) foi configurado com valor igual a 0. O valor padrão de aptidão (parâmetro Q_d) foi configurado com valor igual a 0.1. Foram definidas quatorze respostas de forma que o robô pudesse olhar para o ser humano ou procurar brinquedos em seis regiões definidas ao girar sua cabeça para a esquerda e seis regiões definidas ao girar a sua cabeça para a direita. Isto foi feito para discretizar o ambiente em regiões de interesse que tornaram possível ao robô aprender a seguir o olhar do ser humano para locais corretos, mesmo na presença de objetos distratores. Vale ressaltar que esta discretização foi empregada em diversos trabalhos apresentados por Triesch e seus colegas (Fasel et al., 2002) (Carlson and Triesch, 2003) (Lau and Triesch, 2004) (Deák and Triesch, 2005).

O sistema motivacional foi configurado como a seguir. Foram criadas duas unidades de necessidade: *socialize* e *play*. O limiar de ativação do sistema motivacional foi fixado em 0.70. A inclinação da função sigmóide das unidades de necessidade (parâmetro δ) foi configurada com valor igual a 0.20. Para a unidade *socialize*, o *bias* foi configurado com valor igual a 1.00 e o peso de sua conexão foi configurado com valor igual a 0.5. O peso da conexão recorrente foi configurado com valor igual a 1.00. Os pesos das conexões das unidades de entrada (*hear(attention)*, *see(frontal(face))*, *see(toy(object))*, *see(looking_toy(object))*) foram configurados, respectivamente, com valores iguais a -1.00 , 0.05, 0.05 e 0.00. Para a unidade *play*, o *bias* foi configurado com valor igual a 0.90 e o peso de sua conexão configurado com valor igual a 0.5. O peso da conexão recorrente foi configurado com valor igual a 1.00. Os pesos das conexões das unidades de entrada (*hear(attention)*, *see(frontal(face))*, *see(toy(object))*, *see(looking_toy(object))*) foram configurados, respectivamente, com valores iguais a 0.10, 0.05, 0.05 e -1.00 . Durante o processo de configuração da arquitetura, verificou-se empiricamente que estes valores produziram os melhores resultados.

Conforme a metodologia apresentada na Seção 6.4, estes experimentos fo-

ram compostos por uma fase de aprendizagem de 10.000 unidades de tempo (10.000 segundos no simulador). Durante a fase de aprendizagem, o ser humano mantinha o foco inicialmente no robô até que este estabelecesse o contato ocular com aquele, definido como 3 unidades de tempo mantendo-se o contato ocular. Então, dois objetos eram posicionados no ambiente e o ser humano direcionava o seu olhar para um destes objetos, obedecendo as probabilidades definidas no simulador de interações sociais. O humano mantinha o seu olhar no objeto selecionado por 5 unidades de tempo. Depois, os objetos eram removidos do ambiente social e o ser humano voltava a olhar para o robô, aguardando que este estabelecesse contato ocular novamente. Este procedimento foi executado para simular uma interação social na qual dois agentes estão mantendo contato ocular e então um deles direciona o olhar para um evento ou objeto interessante no ambiente.

Nas primeiras 100 unidades de tempo da fase aprendizagem, nenhum objeto foi posicionado no ambiente e o ser humano manteve o seu foco no robô ao longo deste período. Nestas primeiras 100 unidades de tempo, o robô aprendeu a que manter contato ocular com ser humano, quando ele está olhando para o robô, produz alguns estímulos reforçadores de atenção do ser humano, satisfazendo sua necessidade de socialização (unidade de necessidade *socialize* configurada no sistema motivacional). Este procedimento foi feito para modelar o comportamento do robô de procurar por um ser humano e manter contato ocular sempre que sente necessidades de socialização. Depois das 100 primeiras unidades de tempo, a fase de aprendizagem prosseguiu empregando dois objetos como declarado anteriormente. A partir desta etapa da aprendizagem, o robô sempre olhava para o ser humano quando queria interagir socialmente (unidade de necessidade *socialize* estava ativa). Entretanto, quando um objeto era posicionado no ambiente e o ser humano direcionava seu olhar para este objeto, o robô perdia a atenção do humano e começava a buscar qualquer estímulo no ambiente que pudesse satisfazer seus estados internos (unidades de necessidade *socialize* ou *play*). Quando o robô olhava para um objeto que estava definido como um brinquedo, este satisfazia sua necessidade de brincar, por meio da diminuição do valor de ativação da unidade de necessidade *play*. Então o robô aprendeu que procurar brinquedos no ambiente sempre que tinha a necessidade de brincar.

Adicionalmente, se o robô procurasse por um brinquedo que também era o foco de atenção do ser humano, este voltava a dar atenção ao robô, pela simulação de verbalizações do ser humano em relação ao objeto o qual ambos estavam focando, representado no simulador pelo estímulo *attention*. Deste modo, após de uma história de reforçamento, o robô aprendeu seguir o olhar do ser humano para receber a atenção dele e satisfazer suas necessidades de

brincar e socializar.

A capacidade de aprendizagem da arquitetura foi analisada pela computação da métrica CGI, dada pela Equação (6.1), apresentada na Seção 6.4. Esta métrica é baseada em uma medida proposta por Whalen (Whalen and Schreibman, 2003) e é definida como a frequência de alternância do olhar, a partir do contato ocular com o humano, para um objeto que é foco de atenção do ser humano.

Conforme a metodologia apresentada na Seção 6.4, para quantificar o aprendizado da arquitetura durante os experimentos, a fase de aprendizagem era interrompida em pontos específicos (a cada 500 unidades de tempo) e uma fase de validação do conhecimento adquirido era iniciada para avaliar o comportamento da arquitetura. Esta avaliação era executada por 20 corridas de 500 unidades de tempo (500 ciclos de interação). Para cada corrida, o valor do CGI era computado e, depois das 20 corridas, a média e desvio padrão das 20 medidas eram calculados, dados respectivamente pelas Equações (6.3) e (6.4), apresentadas na Seção 6.4. Depois das 20 corridas da fase de avaliação, a fase de aprendizado era retomada do ponto no qual esta havia sido interrompida.

Na Figura 6.20, é apresentada a curva de aprendizado que demonstra o progresso da aprendizagem durante os experimentos de simulação. Esta figura mostra um gráfico que apresenta o valor da média do CGI para cada fase de avaliação, em pontos específicos durante o processo de aprendizagem. Os resultados obtidos mostram a influência da constante de aprendizagem (parâmetro λ) no desempenho da arquitetura. Os resultados obtidos mostram também que o valor do CGI é crescente no decorrer da fase de aprendizagem, demonstrando as capacidades de aprendizagem da arquitetura.

Adicionalmente ao gráfico contendo a curva de aprendizado, na Tabela 6.3 são apresentados os melhores resultados obtidos empregando os diversos valores da constante de aprendizagem utilizados nos experimentos. Ela mostra a melhor média e desvio padrão obtidos com as avaliações efetuadas durante a fase de aprendizagem, para cada configuração da constante de aprendizagem.

Tabela 6.3: Melhores resultados obtidos durante o processo de aprendizagem.

λ	Melhor média e desvio padrão do CGI
0.9	0.93 ± 0.02
0.7	0.91 ± 0.02
0.5	0.87 ± 0.02
0.3	0.84 ± 0.02
0.1	0.81 ± 0.02

Os resultados mostram que a constante de aprendizagem pode influenciar no desempenho da arquitetura e que o uso de $\lambda = 0.9$ produziu resulta-

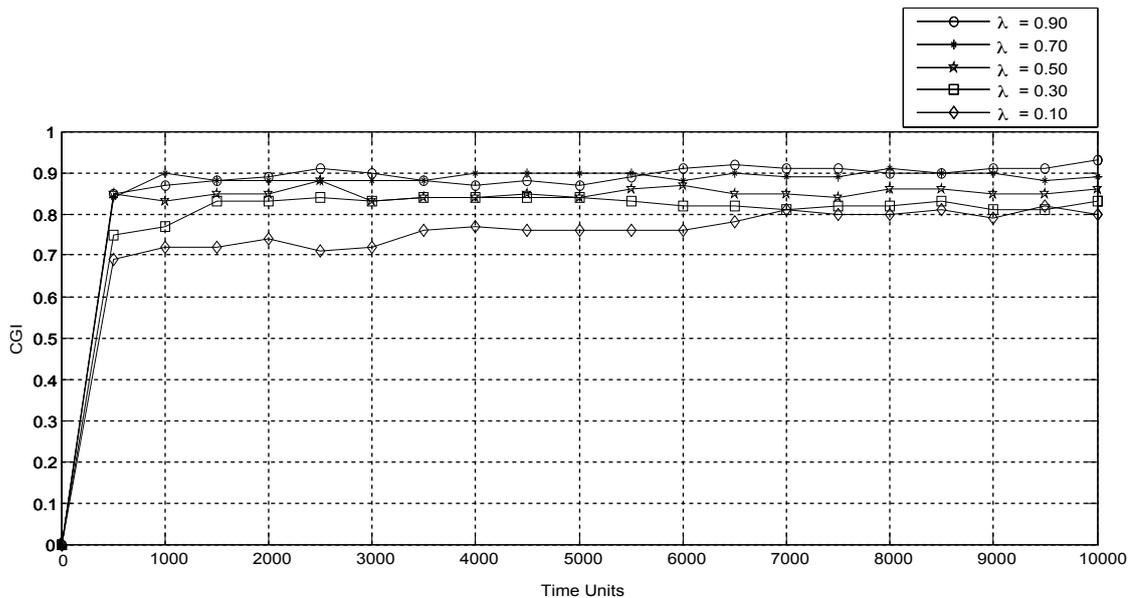


Figura 6.20: Evolução do aprendizado durante os experimentos. As curvas apresentam a avaliação do conhecimento da arquitetura para valores diferentes da constante de aprendizagem (parâmetro λ). Em pontos específicos do processo de aprendizagem, este era temporariamente interrompido e uma fase de avaliação do comportamento do robô era executada por 20 corridas de 500 unidades de tempo. Para cada corrida, era computado o valor do CGI e depois das 20 corridas era calculada a média e desvio padrão das 20 corridas.

dos melhores. Para confirmar estes resultados, foi utilizado o teste *t* pareado para procedimentos bilaterais com 99% de certeza (Moses, 1986) (Mitchell, 1997b). Os resultados são mostrados na Tabela 6.4. Adicionalmente, os resultados mostram que o algoritmo de aprendizado da arquitetura apresenta boa estabilidade, como pode ser visto pelos valores de desvio padrão dos melhores resultados obtidos ($sd = 0.02$).

Tabela 6.4: Resultados do Teste *t* as melhores médias do CGI.

Modelos Comparados	Conclusão
Arquitetura com $\lambda = 0.9$ e Arquitetura com $\lambda = 0.7$	Arquitetura com $\lambda = 0.9$ é superior
Arquitetura com $\lambda = 0.9$ e Arquitetura com $\lambda = 0.5$	Arquitetura com $\lambda = 0.9$ é superior
Arquitetura com $\lambda = 0.9$ e Arquitetura com $\lambda = 0.3$	Arquitetura com $\lambda = 0.9$ é superior
Arquitetura com $\lambda = 0.9$ e Arquitetura com $\lambda = 0.1$	Arquitetura com $\lambda = 0.9$ é superior

Estes resultados também mostram que a arquitetura é capaz de simular os principais processos básicos de aprendizagem evidenciados na Análise do Comportamento. Os resultados demonstram também que a arquitetura é capaz de exibir comportamentos apropriados durante uma interação social. Adicionalmente, os resultados mostram que a arquitetura pode aprender a partir

de uma interação social. Os resultados mostram, ainda, que a arquitetura suporta a modelagem de comportamentos, isso é, ela possibilita que o comportamento do robô seja modelado por aproximações sucessivas, por meio do reforço e encadeamento de comportamentos inatos para a implantação de um comportamento mais complexo. Nestes experimentos, isto foi realizado pela modelagem do comportamento de procurar por um ser humano e, após isso, pela modelagem do comportamento de seguir o olhar deste. A modelagem e encadeamento do comportamento são procedimentos importantes evidenciados na Análise do Comportamento, e são empregados para favorecer a aprendizagem dos comportamentos de um organismo. Portanto, os resultados mostram que a arquitetura é uma ferramenta potencial para controlar robôs sociáveis durante interações em um ambiente social.

Finalmente, um exame dos resultados obtidos mostram que a arquitetura proposta é capaz de representar um modelo de aprendizado da atenção compartilhada e pode-se afirmar que os resultados obtidos estão de acordo com a literatura existente na área (Fasel et al., 2002) (Carlson and Triesch, 2003) (Nagai et al., 2003a) (Nagai et al., 2003b) (Lau and Triesch, 2004) (Deák and Triesch, 2005) (Dube et al., 2004).

Como resultado destes experimentos, foram criadas algumas 293 regras de comportamento. Algumas regras de comportamento foram criadas para controlar o comportamento de procurar e manter o contato ocular com um ser humano:

```

...
at(environment)  $\xrightarrow{0.0249}$  searchhuman()
                 $\xrightarrow{\text{socialize}}$ 
see(frontal(face))  $\xrightarrow{0.2137}$  lookhuman()
                  $\xrightarrow{\text{socialize}}$ 
see(looking_left_r1(face))  $\xrightarrow{0.0003}$  searchhuman()
...

```

As duas primeiras regras controlam o comportamento de procurar por um ser humano sempre que o robô quiser socializar e manter o contato ocular quando eles estiverem olhando um para o outro. A terceira regra controla o comportamento de evitar o contato ocular quando o humano estiver olhando para outro lugar no ambiente, isto pode ser notado pelo baixo valor de aptidão para a execução da resposta (*searchhuman()*). Algumas regras de comportamento foram criadas para controlar o encadeamento do comportamento de estabelecer o contato ocular e posteriormente seguir o olhar do ser humano:

...

$$\text{see}(\text{looking_right_r1}(\text{face})), \text{see}(\text{looking}(\text{face})) \xrightarrow[0.1943]{\text{socialize,play}} \text{searchright}(\text{r2})$$

$$\text{see}(\text{looking_right_r1}(\text{face})), \text{see}(\text{looking}(\text{face})) \xrightarrow{0.0019} \text{searchright}(\text{r3})$$
 ...

A primeira regra controla o comportamento de procurar por um brinquedo em determinada região do ambiente social, uma vez que ele está olhando para o ser humano, mas o ser humano está olhando para outro lugar. Esta regra relaciona a pose estimada da cabeça do ser humano com a região na qual o robô deve procurar pelo brinquedo. Neste caso, quando o ser humano está olhando para a região $r1$ (em relação ao referencial do ser humano) o robô deve procurar por um brinquedo na região $r2$ (em relação ao seu referencial).

A segunda regra é similar à primeira. Entretanto, esta regra relaciona a pose estimada da cabeça do ser humano a uma região na qual o robô não deve procurar por um brinquedo, como pode ser notado pelo baixo valor de aptidão da mesma, ou seja, a execução desta regra de comportamento não resulta em uma consequência agradável para o robô.

Após os primeiros experimentos empregando o simulador de interações sociais, foi realizado um estudo sobre a generalização do conhecimento na forma da representação relacional e, como resultado deste estudo, foi implementado um novo mecanismo de memória de regras de comportamento, que permite a generalização das regras aprendidas pela arquitetura. Este mecanismo, denominado MRC, é apresentado na Seção 6.6 deste Capítulo. A construção deste mecanismo de generalização do conhecimento foi necessária para validar uma das hipóteses de pesquisa deste projeto, em relação à exploração da representação relacional do conhecimento, uma vez que a generalização é uma importante questão de pesquisa na área do aprendizado por reforço relacional (e da área do aprendizado, de um modo geral), e é normalmente realizada empregando-se mecanismos formados por árvores de regressão (Dzeroski et al., 2001) (Driessens, 2004) (Otterlo, 2005). Após esta modificação, foram realizados novos experimentos para avaliar o desempenho desta última versão da arquitetura proposta. Estes experimentos são descritos a seguir.

Este novo conjunto de experimentos seguiu o mesmo processo de simulação descrito no experimento anterior e foi executado para avaliar o desempenho da nova versão da arquitetura empregando o mecanismo MRC. O desempenho desta versão nova da arquitetura robótica foi comparado ao desempenho da versão prévia, que também empregava a representação relacional de conhecimento, mas não era capaz de generalizar o conhecimento aprendido, como possibilitado pelo MRC. Adicionalmente, o desempenho da versão nova

da arquitetura robótica foi comparado com o desempenho de uma terceira versão da arquitetura proposta, empregando como mecanismo de aprendizagem um algoritmo de aprendizagem-Q clássico (*Q-Learning*) (Sutton and Barto, 1998).

A metodologia dos experimentos foi a mesma descrita para o conjunto de experimentos anterior, com uma fase de aprendizagem de 10.000 unidades tempo e interrupções temporárias do processo de aprendizagem em pontos específicos (a cada 500 unidades de tempo) para a execução de uma fase de validação do conhecimento adquirido pela arquitetura. As fases da avaliação também foram executadas como anteriormente, compostas por 20 corridas de 500 unidades de tempo (500 ciclos de interação). Para cada corrida, o valor do CGI era computado e, depois das 20 corridas, a média e desvio padrão das 20 medidas eram calculados, dados respectivamente pelas Equações (6.3) e (6.4), apresentadas na Seção 6.4. Depois das 20 corridas da fase de avaliação, a fase de aprendizado era retomada do ponto no qual esta havia sido interrompida.

Para este conjunto de experimentos, o conhecimento de arquitetura foi configurado da mesma maneira que anteriormente. Quatro estímulos foram declarados: *face*, *object*, *attention* e *environment*, no qual *attention* é um estímulo reforçador gerado com a atenção do ser humano. Foram declarados dois fatos para definir que objetos vermelhos e azuis são brinquedos. Também foram declarados três fatos para diferenciar a pose da cabeça do ser humano como frontal, além de seis poses de perfil esquerdo e seis poses de perfil direito. Adicionalmente, foram declarados mais fatos para definir quando o robô está focalizando o ser humano ou um brinquedo.

Tanto para a versão prévia da arquitetura como a nova versão, o módulo de emissão de respostas foi configurado como a seguir. A constante de aprendizagem (parâmetro λ) foi configurada com valor igual a 0.9. O número mínimo de execuções das regras de comportamento antes de se iniciar o decréscimo da taxa de aprendizagem (parâmetro $N_{Interactions}$) foi configurado com valor igual a 0. O valor padrão de aptidão (parâmetro Q_d) foi configurado com valor igual a 0.1. Foram definidas quatorze respostas de forma que o robô pudesse olhar para o ser humano ou procurar brinquedos em seis regiões definidas ao girar sua cabeça para a esquerda e seis regiões definidas ao girar a sua cabeça para a direita. Isto foi feito para dividir o ambiente em regiões de interesse que tornaram possível ao robô aprender a seguir o olhar do ser humano para locais corretos, mesmo na presença de objetos distratores.

O sistema motivacional foi configurado como a seguir. Foram criadas duas unidades de necessidade: *socialize* e *play*. O limiar de ativação do sistema motivacional foi fixado em 0.70. A inclinação da função sigmóide das unidades de necessidade (parâmetro δ) foi configurada com valor igual a 0.20. Para a

unidade *socialize*, o *bias* foi configurado com valor igual a 1.00 e o peso de sua conexão foi configurado com valor igual a 0.5. O peso da conexão recorrente foi configurado com valor igual a 1.00. Os pesos das conexões das unidades de entrada (*hear(attention)*, *see(frontal(face))*, *see(toy(object))*, *see(looking_toy(object))*) foram configurados, respectivamente, com valores iguais a -1.00 , 0.05 , 0.05 e 0.00 . Para a unidade *play*, o *bias* foi configurado com valor igual a 0.90, e o peso de sua conexão configurado com valor igual a 0.5. O peso da conexão recorrente foi configurado com valor igual a 1.00. Os pesos das conexões das unidades de entrada (*hear(attention)*, *see(frontal(face))*, *see(toy(object))*, *see(looking_toy(object))*) foram configurados, respectivamente, com valores iguais a 0.10 , 0.05 , 0.05 e -1.00 .

Na nova versão da arquitetura, a rede neural ART2 foi configurada como descrito a seguir. O parâmetro de vigilância (λ) foi configurado com valor igual a 0.95. Os parâmetros a , b , c e d foram configurados respectivamente com valores iguais a 10.0, 10.0, 0.1 e 0.9. Na versão da arquitetura que emprega o *Q-Learning*, o algoritmo de aprendizado foi configurado como a seguir. O parâmetro de exploração e reaproveitamento (exploitation/exploitation) (ϵ) foi configurado com valor igual a 0.5%. A taxa de aprendizagem (α) foi configurada com valor igual a 0.2. A taxa de desconto (γ) foi configurada com valor igual a 0.1.

Na Figura 6.21 é apresentada a curva de aprendizado que demonstra o progresso da aprendizagem durante os experimentos de simulação. Esta figura mostra um gráfico que apresenta o valor da média do CGI para cada fase de avaliação, em pontos específicos durante o processo de aprendizagem. As três curvas exibidas no gráfico mostram o desempenho das três versões da arquitetura, comparadas nos experimentos. Os resultados obtidos mostram que o valor do CGI é crescente no decorrer da fase de aprendizagem, demonstrando as capacidades de aprendizagem das três versões da arquitetura.

Adicionalmente ao gráfico contendo a curva de aprendizado, na Tabela 6.5 são apresentados os melhores resultados obtidos empregando os diversos valores da constante de aprendizagem utilizados nos experimentos. Ela mostra a melhor média e desvio padrão obtidos com as avaliações efetuadas durante a fase de aprendizagem, para cada versão da arquitetura.

Tabela 6.5: Melhores resultados obtidos durante o processo de aprendizagem.

Versão da Arquitetura	Melhor média e desvio padrão do CGI
Versão prévia da arquitetura	0.97 ± 0.03
Nova versão da arquitetura	0.96 ± 0.03
Arquitetura com <i>Q-Learning</i>	0.98 ± 0.03

Os resultados mostram que as três versões da arquitetura apresentaram

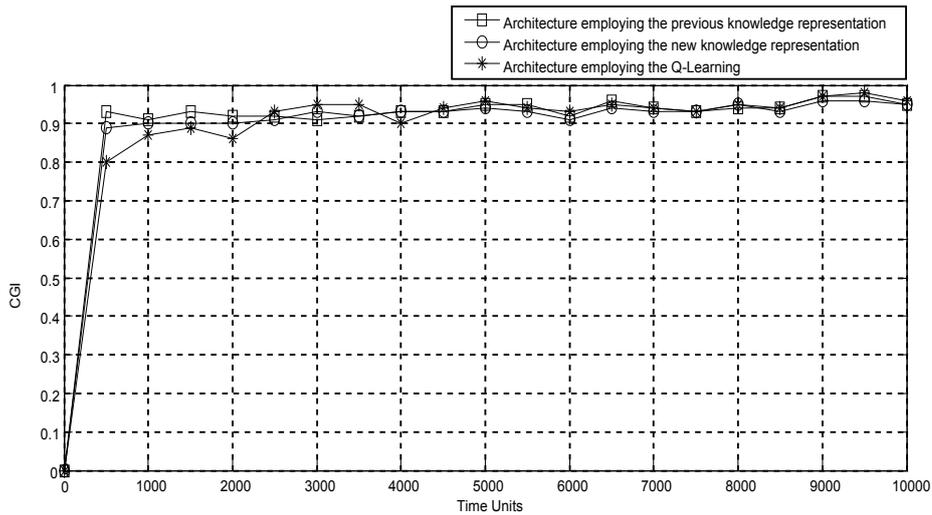


Figura 6.21: Evolução do aprendizado durante os experimentos. As três curvas apresentam a avaliação do conhecimento da arquitetura para cada uma das três versões da arquitetura, comparadas durante os experimentos. Em pontos específicos do processo de aprendizagem, este era temporariamente interrompido e uma fase de avaliação do comportamento do robô era executada por 20 corridas de 500 unidades de tempo. Para cada corrida, era computado o valor do CGI e depois das 20 corridas era calculada a média e desvio padrão das 20 corridas.

desempenhos similares, considerando as melhores médias dos CGIs obtidas durante o processo de aprendizado. Para confirmar estes resultados, foi utilizado o teste *t* pareado para procedimentos bilaterais com 99% de certeza (Moses, 1986) (Mitchell, 1997b). Os resultados são mostrados na Tabela 6.6. Adicionalmente, os resultados mostram que os algoritmos de aprendizagem das três versões da arquitetura apresentam boa estabilidade, como pode ser visto pelos valores de desvio padrão dos melhores resultados obtidos ($sd = 0.03$).

Tabela 6.6: Resultados do Teste *t* as melhores médias do CGI.

Modelos Comparados	Conclusão
Nova versão da arquitetura e versão prévia da arquitetura	Desempenho Similar
Nova versão da arquitetura e arquitetura com <i>Q-Learning</i>	Desempenho Similar
Versão prévia da arquitetura e arquitetura com <i>Q-Learning</i>	Desempenho Similar

Entretanto, os resultados mostram que no início da fase de aprendizagem (até 2.000 unidades de tempo), as versões das arquiteturas empregando a representação relacional (representação prévia e a nova representação híbrida) apresentaram desempenho superior à versão da arquitetura empregando o *Q-Learning* como algoritmo de aprendizagem. Estes resultados são apresentados na Tabela 6.7.

Tabela 6.7: Resultados obtidos em 2.000 unidades de tempo de simulação.

λ	Média e desvio padrão do CGI
Versão prévia da arquitetura	0.92 ± 0.02
Nova versão da arquitetura	0.90 ± 0.01
Arquitetura com <i>Q-Learning</i>	0.86 ± 0.03

Este resultado também foi confirmado por um *teste t* pareado para procedimentos bilaterais com 99% de certeza (Moses, 1986) (Mitchell, 1997b). Os resultados são mostrados na Tabela 6.8.

Tabela 6.8: Resultados do Teste *t* as melhores médias do CGI.

Modelos Comparados	Conclusão
Nova versão da arquitetura e versão prévia da arquitetura	Desempenho Similar
Nova versão da arquitetura e arquitetura com <i>Q-Learning</i>	A nova versão da arquitetura é superior
Versão prévia da arquitetura e arquitetura com <i>Q-Learning</i>	A versão prévia da arquitetura é superior

Estes resultados podem ser explicados considerando-se as argumentações dos principais pesquisadores da área do aprendizado por reforço relacional (Dzeroski et al., 2001) (Driessens, 2004) (Otterlo, 2005), que defendem a superioridade da representação relacional sobre a representação clássica tabular da função Q , empregada no aprendizado por reforço. Estes pesquisadores também defendem que um algoritmo de aprendizado por reforço com representação tabular pode levar um tempo mais longo para convergir para uma política ótima em grandes domínios de problema. Portanto, esta seria uma possível explicação para o melhor desempenho inicial das versões prévia e nova da arquitetura em relação à versão empregando o *Q-Learning*.

De fato, para estes experimentos, o *Q-Learning* empregou como mecanismo de representação do conhecimento uma *Tabela Q* de 3150 posições. Por outro lado, a versão prévia da arquitetura, empregando a representação relacional do conhecimento, aprendeu 345 regras de comportamento durante a fase de aprendizagem e a nova versão da arquitetura, empregando a representação híbrida do conhecimento, aprendeu 285 regras de comportamento durante a fase de aprendizagem. Estes resultados sugerem que a representação relacional, tanto da versão prévia da arquitetura quanto da nova, podem permitir uma representação do conhecimento de um modo mais econômico e mais eficiente. Este resultado também sugere que a nova versão da arquitetura é capaz de generalizar o conhecimento aprendido, uma vez que necessitou de um número menor de regras de comportamento para cumprir a mesma tarefa que a versão prévia da arquitetura.

Os resultados obtidos neste novo conjunto de experimento confirmam os

resultados obtidos anteriormente e mostram que a arquitetura pode exibir comportamentos apropriados durante uma interação social e aprender a partir desta interação, formando a base para a construção de uma ferramenta promissora para controlar robôs sociáveis.

Finalmente, um exame dos resultados obtidos mostram que o mecanismo híbrido de representação do conhecimento pode representar o conhecimento de maneira mais eficaz que uma técnica de AR clássica. Portanto, pode-se afirmar que os resultados obtidos estão de acordo com a literatura existente (Dzeroski et al., 2001) (Driessens, 2004) (Otterlo, 2005) e que o mecanismo proposto (MRC) constitui uma contribuição para a área de pesquisa.

6.9 Cabeça Robótica Interativa

Após a realização dos experimentos sobre o aprendizado da atenção compartilhada, empregando o simulador de interações sociais, foram realizados os últimos experimentos e estudos previstos neste projeto de pesquisa, empregando a cabeça robótica interativa apresentada na Seção 6.5. Estes experimentos foram realizados no contexto do aprendizado da atenção compartilhada com o propósito de se avaliar a capacidade de aprendizagem da arquitetura proposta em um ambiente social real e controlado.

Adicionalmente, para demonstrar o valor do aprendizado da atenção compartilhada como precursor do aprendizado por meio de interações sociais, foi realizado um conjunto de experimentos no contexto do aprendizado por tutoria, empregando o mecanismo de aprendizagem por tutoria proposto neste trabalho, evidenciando que a atenção compartilhada permite o direcionamento da atenção de um agente sociável para o aprendizado de conceitos importantes do mundo real.

6.9.1 Experimentos de Aprendizado da Atenção Compartilhada

Nesta seção, são apresentados os principais resultados dos experimentos executados para a avaliação da arquitetura proposta no domínio de aplicação da aprendizagem da atenção compartilhada, empregando a cabeça robótica interativa. O propósito destes experimentos foi a determinação da capacidade de exibição de comportamentos apropriados e de aprendizagem da arquitetura durante o controle do robô em um ambiente social real e controlado.

Para este novo conjunto de experimentos, o conhecimento de arquitetura foi configurado da seguinte maneira. Quatro estímulos foram declarados: *face*, *object*, *attention* e *environment*, no qual *attention* é um estímulo reforçador gerado com a atenção do ser humano. Foram declarados quatro fatos para definir que objetos vermelhos, amarelos, laranja e verdes são frutas. Também

foram declarados seis fatos para diferenciar a pose da cabeça do ser humano. Adicionalmente, foram definidos alguns fatos para definir quando o robô está focalizando o ser humano ou uma fruta.

O módulo de emissão de respostas foi configurado como a seguir. A constante de aprendizagem (parâmetro λ) foi configurada com valor igual a 0.9. O número mínimo de execuções das regras de comportamento antes de se iniciar o decréscimo da taxa de aprendizagem (parâmetro $N_{Interactions}$) foi configurado com valor igual a 0. O valor padrão de aptidão (parâmetro Q_d) foi configurado com valor igual a 0.05. Foram definidas sete respostas de forma que o robô pudesse olhar para o ser humano ou procurar frutas em cinco regiões definidas ao girar sua cabeça para a esquerda ou para a direita. Isto foi feito para discretizar o ambiente em regiões de interesse que tornaram possível ao robô aprender a seguir o olhar do ser humano para locais corretos, como nos experimentos empregando o simulador de interações sociais.

O sistema motivacional foi configurado como descrito a seguir. Foram criadas duas unidades de necessidade: *socialize* e *play*. O limiar de ativação do sistema motivacional foi fixado em 0.50. A inclinação da função sigmóide das unidades de necessidade (parâmetro δ) foi configurada com valor igual a 0.20. Para a unidade *socialize*, o *bias* foi configurado com valor igual a 1.00 e o peso de sua conexão foi configurado com valor igual a 0.5. O peso da conexão recorrente foi configurado com valor igual a 1.00. Os pesos das conexões das unidades de entrada (*hear(attention)*, *see(frontal(face))*, *see(looking_frontal(face))*, *see(looking_fruit(object))*) foram configurados, respectivamente, com os valores -1.50, 0.95, -1.50 e 0.50. Para a unidade *play*, o *bias* foi configurado com os valores 0.90, e o peso de sua conexão, configurado com valor igual a 0.5. O peso da conexão recorrente foi configurado com valor igual a 1.00. Os pesos das conexões das unidades de entrada (*hear(attention)*, *see(frontal(face))*, *see(looking_frontal(face))*, *see(looking_fruit(object))*) foram configurados, respectivamente, com os valores -0.90, 1.00, 0.90 e -1.50. Durante o processo de configuração da arquitetura, verificou-se empiricamente que estes valores produziram os melhores resultados.

Conforme a metodologia apresentada na Seção 6.4, os experimentos foram compostos por uma fase de aprendizagem de 1.000 ciclos de interação. Durante a fase de aprendizagem, o ser humano mantinha o foco inicialmente no robô até que este estalesse o contato ocular. Então, uma fruta era posicionada no ambiente e o ser humano direcionava o seu olhar a mesma. O ser humano mantinha o seu olhar no objeto selecionado até o robô emitir uma resposta (execução uma ação motora). Depois, a fruta era removida e o ser humano voltava a olhar para o robô, aguardando que este estabelecesse novamente o contato ocular. Este procedimento foi executado para simular

uma interação social na qual dois agentes estão mantendo contato ocular e então um deles direciona o olhar para um evento ou objeto interessante no ambiente.

Nas primeiras 30 unidades de tempo da fase aprendizagem, nenhum objeto foi posicionado no ambiente e o ser humano manteve o seu foco no robô ao longo de todo este período de tempo. Nestas primeiras 30 unidades, o robô aprendeu que estabelecer o contato ocular com ser humano produz alguns estímulos reforçadores de atenção do mesmo, satisfazendo sua necessidade de socialização (unidade de necessidade *socialize* configurada no sistema motivacional). Este procedimento foi executado para modelar o comportamento do robô de procurar por um ser humano e manter contato ocular sempre que sente necessidades de socialização. Depois das 30 primeiras unidades de tempo, a fase de aprendizagem prosseguiu empregando uma fruta como declarado acima. A partir deste ponto da aprendizagem, o robô sempre olhava para o ser humano quando queria interagir socialmente (unidade de necessidade *socialize* estava ativa). Entretanto, quando uma fruta era posicionada no ambiente e o ser humano direcionava seu olhar para esta, o robô perdia a atenção do humano e começava a procurar por qualquer estímulo no ambiente que pudesse satisfazer seus estados internos (unidades de necessidade *socialize* ou *play*). Quando o robô olhava para um objeto definido como uma fruta na base de conhecimento da arquitetura, este satisfazia suas necessidades, por meio da diminuição do valor de ativação da unidade de necessidade *play*. Adicionalmente, se o robô focasse uma fruta que também era o foco de atenção do ser humano, este último voltava a dar atenção ao robô, em relação a fruta a qual ambos estavam focando.

Como as frutas eram posicionadas no ambiente somente quando o ser humano direcionava seu foco para as mesmas, depois de uma história de reforçamento o robô aprendeu seguir o olhar do ser humano para receber a atenção dele e satisfazer suas necessidades de brincar e socializar.

A capacidade de aprendizagem da arquitetura foi analisada observando-se a interação do robô com o ser humano e com o ambiente e pela computação da métrica CGI), dada pela Equação (6.1), apresentada na Seção 6.4. Conforme a metodologia apresentada na Seção 6.4, para quantificar o aprendizado da arquitetura durante os experimentos, a fase de aprendizagem era interrompida em pontos específicos (a cada 100 unidades de tempo) e uma fase de validação do conhecimento adquirido era iniciada para avaliar o comportamento da arquitetura. Esta avaliação era executada por 20 corridas de 100 unidades de tempo (100 ciclos de interação). Para cada corrida, o valor do CGI era computado e, depois das 20 corridas, a média e desvio padrão das 20 medidas eram calculados, dados respectivamente pelas Equações (6.3) e (6.4), apre-

sentadas na Seção 6.4. Depois das 20 corridas da fase de avaliação, a fase de aprendizado era retomada do ponto no qual esta havia sido interrompida.

Na Figura 6.22, é apresentada a curva de aprendizado que demonstra o progresso da aprendizagem durante os experimentos. Esta figura mostra um gráfico que apresenta o valor da média do CGI para cada fase de avaliação, em pontos específicos durante a fase de aprendizagem. Os resultados obtidos mostram que o valor do CGI é crescente no decorrer da fase de aprendizagem, demonstrando as capacidades de aprendizagem da arquitetura.

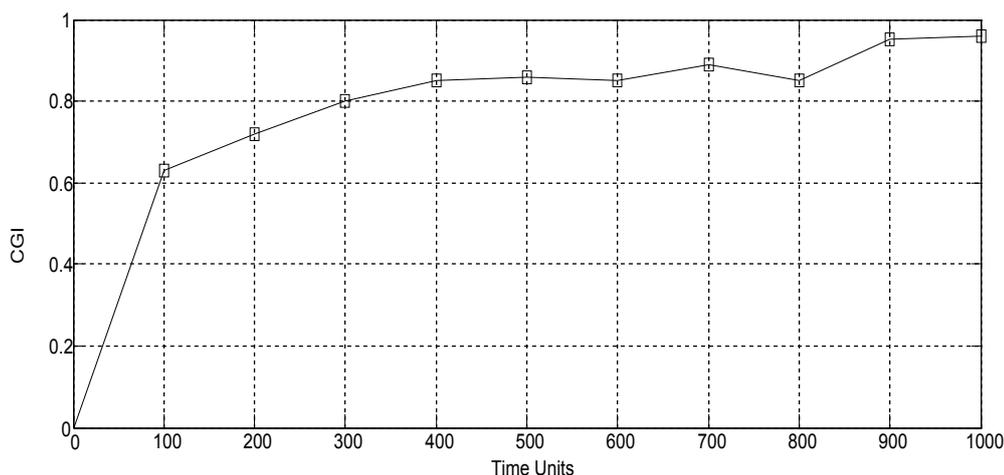


Figura 6.22: Evolução do aprendizado durante os experimentos. A curva apresenta a avaliação do conhecimento adquirido pela arquitetura com o passar do tempo. Em pontos específicos do processo de aprendizagem, este era temporariamente interrompido e uma fase de avaliação do comportamento do robô era executada por 20 corridas de 100 unidades de tempo. Para cada corrida, era computado o valor do CGI e depois das 20 corridas era calculada a média e desvio padrão das 20 corridas.

Estes resultados mostram que a arquitetura é capaz de exibir comportamentos apropriados durante uma interação social real e controlada. Adicionalmente, os resultados confirmam os resultados dos experimentos anteriores, mostrando que a arquitetura pode aprender a partir de uma interação social. Os resultados mostram ainda que a arquitetura suporta a modelagem de comportamentos, isso é, ela possibilita que o comportamento do robô seja modelado por meio de aproximações sucessivas, por meio do reforço e encadeamento de comportamentos inatos para a implantação de um comportamento mais complexo. Nestes experimentos, isto foi realizado pela modelagem do comportamento de procurar por um ser humano e, após isso, pela modelagem do comportamento de seguir o olhar deste.

Finalmente, um exame dos resultados obtidos mostram que a arquitetura proposta é capaz de adquirir habilidades sociais básicas a partir de comportamentos inatos existentes no repertório do robô e da interação com o ambiente.

Os resultados evidenciam também que a arquitetura constitui uma contribuição para a área de pesquisa sobre o aprendizado da atenção compartilhada, por representar um modelo de aprendizado desta habilidade tida como um dos grandes desafios da robótica sociável. Adicionalmente, pode-se afirmar que os resultados obtidos estão de acordo com a literatura existente na área (Fasel et al., 2002) (Carlson and Triesch, 2003) (Nagai et al., 2003a) (Nagai et al., 2003b) (Lau and Triesch, 2004) (Deák and Triesch, 2005) (Dube et al., 2004).

Portanto, os resultados mostram que a arquitetura é uma ferramenta potencial para controlar robôs sociáveis durante interações em um ambiente social.

6.9.2 Experimentos de Aprendizado por Tutelagem

Nesta seção são apresentados e discutidos os principais resultados obtidos com os experimentos executados para se avaliar a interface multimodal e o mecanismo de aprendizagem por tutelagem proposto nesta pesquisa. No cenário experimental, um ser humano direcionava a atenção do robô e lhe apresentava diversos objetos com o objetivo de ensinar os nomes dos mesmos ao robô. Como apontado anteriormente, este experimento demonstra o valor do aprendizado da atenção compartilhada como precursor do aprendizado social. Os objetos empregados nos experimentos foram 4 tipos de frutas: uma *maçã vermelha*, um *limão amarelo*, uma *laranja*, e uma *romã avermelhada*. O propósito dos experimentos foi a avaliação da capacidade do mecanismo de aprendizagem em exibir comportamento social apropriado, em aprender a partir da interação social e generalizar os conceitos aprendidos sobre os objetos. Na Figura 6.23, é ilustrado o processamento de imagem executado pelo sistema de visão quando uma maçã foi apresentada ao robô.

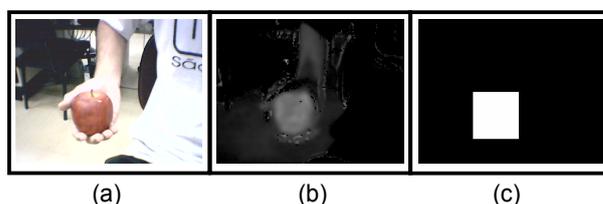


Figura 6.23: Processamento de imagem executado pelo sistema de visão quando uma maçã foi apresentada ao robô (a). Primeiro, o sistema de visão processa o mapa de saliência (b). Então, o sistema seleciona uma área de interesse (c) baseado em dois limiares: saliência e raio mínimo da região.

Para os experimentos, o mecanismo de aprendizagem por tutelagem foi configurado da seguinte maneira. O *limiar de confiança* (ϕ_c) foi configurado com valor igual a 2 e o *limiar de conhecimento* (ϕ_k) foi configurado com valor

igual 6. A rede neural de ART2 foi configurada como a seguir. O parâmetro de vigilância (λ) foi configurado com valor igual 0.999. Os parâmetros a , b , c e d foram configurados respectivamente com valores iguais 10.0, 10.0, 0.1 e 0.9. Durante o processo de configuração do mecanismo, verificou-se empiricamente que estes valores produziram os melhores resultados.

Adicionalmente, a arquitetura foi configurada com o conhecimento prévio que a permitiu ativar o mecanismo de aprendizagem por tutelação quando o ser humano solicitava ao robô o reconhecimento do objeto apresentado. Este conhecimento prévio foi configurado na forma de estímulos auditivos que codificavam os diálogos previstos durante os experimentos. Adicionalmente, o conhecimento foi configurado na forma de regras de comportamento como a regra a seguir:

```

...
see(looking_fruit(object))&hear(speech_identify)  $\xrightarrow[\text{socialize}]{1.00}$  recognizeobject()
...

```

Os experimentos foram compostos por uma fase de apresentação na qual as 4 frutas foram apresentadas sob 5 diferentes condições de iluminação: *todas as luzes acesas*, *somente a iluminação natural*, *iluminação natural com uma fonte de luz diretamente acima da fruta*, *iluminação natural com uma fonte de luz acima e à esquerda da fruta*, *iluminação natural com uma fonte de luz acima e à direita da fruta*, resultando em 20 apresentações das frutas para cada fase de apresentação.

Para avaliar o mecanismo de aprendizagem proposto, foram calculadas 5 métricas durante os experimentos: *taxa de conhecimento*, *taxa de suposições corretas*, *taxa de suposições incorretas*, *taxa de erros* e *taxa de acertos*. A *taxa de conhecimento* é a frequência na qual o mecanismo de aprendizagem entrou no modo de *conhecimento*. A *taxa de suposições incorretas* é a frequência na qual o mecanismo de aprendizagem entrou no modo de *incerteza* e supôs incorretamente o nome da fruta. A *taxa de suposições corretas* é a frequência na qual o mecanismo de aprendizagem entrou no modo de *incerteza* e supôs corretamente o nome da fruta. A *taxa de erros* é a frequência na qual o mecanismo de aprendizagem entrou no modo de *conhecimento*, mas apontou incorretamente o nome da fruta. A *taxa de acertos* é a frequência na qual o mecanismo de aprendizagem entrou no modo de *conhecimento* e apontou corretamente o nome da fruta.

Para quantificar a capacidade de aprendizagem do mecanismo proposto, a fase de apresentação foi repetida 20 vezes (20 corridas), variando-se a seqüência das condições de iluminação, sempre esvaziando a memória de conceitos no início de uma nova corrida. Depois de cada corrida, foram calculadas as 5

métricas descritas anteriormente e estas foram armazenadas. Então, depois das 20 corridas, a média e o desvio padrão das métricas foi calculado.

Na Tabela 6.9 são mostrados os valores das médias e desvios padrão das 5 métricas para as 20 corridas executadas durante os experimentos. Adicionalmente, na Tabela 6.9 é mostrada a média e o desvio padrão do tamanho da memória de conceitos ao término de cada corrida, exibindo o número de novos conceitos aprendido na fase de apresentação.

Tabela 6.9: Resultados obtidos após as 20 corridas de sessão de aprendizado por tutelação.

Métrica	Média (%)
Taxa de conhecimento	7.50 ± 4.93
Taxa de suposições corretas	15.00 ± 7.34
Taxa de suposições incorretas	1.25 ± 2.64
Taxa de erros	1.88 ± 2.03
Taxa de acertos	74.38 ± 7.48
Tamanho da memória de conceitos	8 ± 1

Os resultados mostram que o mecanismo de aprendizagem é capaz de exibir comportamento apropriado e aprender a partir de interações sociais. Os resultados mostram também que durante os experimentos o mecanismo de aprendizagem foi capaz de exibir *conhecimento*, *incerteza*, e *certeza* sobre os nomes das frutas, durante a fase de apresentação, permitindo um processo de aprendizagem socialmente direcionado e de uma maneira mais natural. O uso de certeza e incerteza sobre um objeto permitiu ao ser humano determinar a compreensão exata do robô sobre o conceito aprendido. Os exemplos seguintes ilustram como um ser humano interage com o robô de acordo com suas respostas:

[Ser humano apresenta uma maçã vermelha]

h -- Robô, o que é isso?

r -- Eu não sei!

h -- Isto é uma maçã.

r -- Certo.

h -- Isto mesmo!

...

[A condição de iluminação é alterada]

[Ser humano apresenta uma maçã vermelha]

h -- Robô, o que é isso?

r -- Parece uma maçã.

h -- Muito bom!

...

[A condição de iluminação é alterada novamente]
[Ser humano apresenta uma maçã vermelha]
h -- Robô, o que é isso?
r -- Isto é uma maçã
h -- Muito bom!
...

Os experimentos demonstraram como é possível transformar um problema de aprendizado de máquina em um problema de colaboração entre robôs e seres humanos, empregando as habilidades sociais naturais dos seres humanos para ensinar um robô.

Os resultados também mostram que o mecanismo de aprendizagem pode generalizar o conhecimento instruído, uma vez que o tamanho médio da memória de conceitos é 8, apesar da apresentação de 20 conceitos nas fases de apresentação. Esta generalização é executada, no mecanismo de aprendizagem, por meio do agrupamento incremental de conceitos similares pela rede neural ART2.

Finalmente, um exame dos resultados obtidos mostram que a arquitetura proposta é capaz controlar um robô sociável em uma interação controlada, ainda que simples em comparação com muitos trabalhos existentes na literatura. Adicionalmente, os resultados evidenciam que a atenção compartilhada é uma habilidade social fundamental para o desenvolvimento social. Os resultados mostram também como um diálogo colaborativo pode permitir que um robô aprenda conceitos sobre objetos ou eventos importantes do ambiente. Vale ressaltar aqui que os resultados obtidos estão de acordo com a literatura existente (Nagai et al., 2003a) (Nagai et al., 2003b) (Dube et al., 2004) (Lockerd and Breazeal, 2004).

Portanto, pode-se concluir que o mecanismo de aprendizagem por tutela-gem, assim como a arquitetura robótica, constituem uma ferramenta promissora para controlar robôs sociáveis durante interações em um ambiente social real e controlado.

6.10 Considerações Finais

Este Capítulo apresentou a arquitetura robótica proposta neste trabalho de pesquisa. A arquitetura proposta foi avaliada em diversos experimentos executados em diferentes domínios de aplicação, demonstrando a portabilidade da arquitetura.

Os resultados obtidos com os experimentos demonstram a potencialidade da arquitetura e dos demais módulos desenvolvidos como ferramenta para a

construção de robôs sociáveis para diversos domínios de problema. Os resultados obtidos mostram que a arquitetura é capaz de exibir comportamentos apropriados e é capaz de aprender a partir de interações sociais.

Tais resultados também mostram que a forma de representação híbrida do conhecimento, adotada para a arquitetura, é capaz de representar grandes quantidades de conhecimento de maneira econômica e é capaz de generalizar de maneira incremental o conhecimento adquirido. Adicionalmente, esta forma de representação pode ser mais natural para um especialista em robótica durante a configuração da arquitetura.

Ainda, estes resultados mostram que o mecanismo de aprendizagem é capaz de exibir comportamento apropriado e aprender a partir de interações sociais. Os resultados mostram também que, durante os experimentos, este mecanismo permite um processo de aprendizagem socialmente direcionado e de uma maneira mais natural.

No próximo Capítulo é apresentada uma discussão geral sobre esta pesquisa. Adicionalmente, são feitas as considerações finais deste trabalho.

Considerações Finais

O propósito desta pesquisa foi a exploração e o desenvolvimento de estruturas e mecanismos que permitam o controle de robôs sociáveis. Um propósito adicional foi a organização destas estruturas e métodos em uma arquitetura robótica com componentes reaproveitáveis, que possam ser utilizados para a construção de robôs sociáveis para diversas aplicações. Ainda, o propósito deste trabalho foi a investigação de formas alternativas de representação do conhecimento capazes de representar e manipular grandes espaços de busca e decisão.

Um exame da arquitetura proposta mostra que ela contém todos os elementos-chaves das principais arquiteturas de controle de robôs sociáveis encontradas na literatura, constituindo uma importante contribuição para a literatura da área de pesquisa. O sistema motivacional proposto, embora simples, é capaz de simular as necessidades internas do robô, podendo adicionar pró-atividade ao comportamento social exibido durante as interações com o ambiente. Uma deficiência que deve ser apontada em relação ao sistema motivacional proposto é a necessidade de se configurar os parâmetros do mesmo de forma empírica e manual. Esta configuração pode se tornar uma tarefa complexa dependendo do domínio da aplicação para o qual a arquitetura está sendo configurada. O sistema de percepção da arquitetura suporta a conexão de diversos módulos externos de percepção como visão computacional, voz e algoritmos de aquisição de dados. Dentro dos limites da estrutura da arquitetura, o sistema de percepção serve, principalmente, ao propósito de codificar os estímulos recebidos dos módulos de percepção externos e propagá-los para os demais módulos da arquitetura. O sistema de comportamento da arquitetura proposta é integrado ao sistema de aprendizagem formado pelo algoritmo de aprendizado

por reforço não determinístico. Este sistema constitui uma importante contribuição desta pesquisa, uma vez que algumas das principais arquiteturas de controle de robôs sociáveis encontradas na literatura não suportam o aprendizado. Vale ressaltar aqui que o sistema de aprendizagem é capaz de simular os processos básicos de aprendizagem evidenciados na Análise do Comportamento. Este sistema se diferencia da maioria dos sistemas existentes por constituir um sistema de aprendizagem e seleção por conseqüências.

Outra importante contribuição deste trabalho é a exploração de novas formas da representação do conhecimento. Como pode-se verificar nos resultados obtidos com os experimentos executados, a forma de representação relacional pode possibilitar uma representação mais eficiente de grandes espaços de busca e decisão. Os resultados obtidos nos experimentos comparativos entre a arquitetura proposta empregando o MRC e o algoritmo *Q-Learning* clássico estão de acordo com os resultados da literatura que abordam uma importante questão de pesquisa da área do aprendizado por reforço relacional, a generalização incremental do conhecimento adquirido (Dzeroski et al., 2001) (Driessens, 2004) (Otterlo, 2005). Ainda, deve-se ressaltar aqui que o conhecimento codificado na forma de representação adotada pode ser facilmente interpretado por um especialista durante a configuração da arquitetura e pode ser facilmente complementado por conhecimento prévio do domínio da aplicação. Este é um importante requisito para um arquitetura robótica que precisa ser configurada para ser reutilizada em diversos domínios de aplicação.

O propósito comum a todos os experimentos realizados durante este trabalho de pesquisa foi a validação da capacidade de aprendizagem e controle da arquitetura proposta durante as interações sociais. O objetivo final destes experimentos é a validação da arquitetura aplicada ao controle de um robô real em um ambiente social controlado.

Os experimentos executados empregando o simulador de interações sociais demonstraram que o desenvolvimento de uma arquitetura robótica constituída por mecanismos e estruturas inspiradas na análise do comportamento pode ser uma ferramenta promissora para a construção de robôs sociáveis. Os experimentos demonstraram também que a forma de representação do conhecimento adotada pode representar grandes espaços de busca de maneira mais econômica e eficiente, constituindo um importante requisito para a aplicação da arquitetura em problemas reais. Adicionalmente, o aprendizado da atenção visual compartilhada, realizado nos experimentos executados, constitui uma importante contribuição para a área de pesquisa da robótica sociável, uma vez que esta habilidade é apontada como fundamental e um grande desafio desta área de pesquisa. Vale destacar aqui que os experimentos seguiram uma metodologia similar à utilizada nos trabalhos apresentados por Triesch e

seus colegas (Fasel et al., 2002) (Carlson and Triesch, 2003) (Lau and Triesch, 2004) (Deák and Triesch, 2005) e que os resultados obtidos estão de acordo com os demais trabalhos existentes na área (Nagai et al., 2003a) (Nagai et al., 2003b) (Dube et al., 2004).

Os experimentos executados empregando a cabeça robótica interativa confirmaram os resultados obtidos com simulador de interações sociais. Vale ressaltar aqui que a arquitetura robótica constituída por mecanismos e estruturas inspiradas na análise do comportamento pode ser uma ferramenta promissora para a construção de robôs sociáveis reais. Os experimentos demonstraram também que a forma de representação do conhecimento adotada pode ser promissora para a aplicação da arquitetura em problemas reais. Adicionalmente, o aprendizado da atenção visual compartilhada em um ambiente social real, demonstrado nos experimentos executados, demonstram que a arquitetura constitui um modelo promissor para o aprendizado de habilidades sociais básicas e constitui importante contribuição para a área de pesquisa da robótica sociável, uma vez que esta habilidade é apontada como fundamental e como um grande desafio desta área de pesquisa. O desenvolvimento do módulo de aprendizagem por tutela e os experimentos realizados demonstram como a arquitetura robótica proposta pode ser facilmente estendida pela integração de novos módulos sociais. Estes experimentos também demonstram o valor do aprendizado da atenção compartilhada como precursor do aprendizado social. Adicionalmente, estes experimentos mostram a capacidade da arquitetura em utilizar o conhecimento previamente adquirido para executar tarefas mais complexas, em conformidade com os ciclos de evolução de projetos apresentados na Seção 2.2.

É importante neste ponto destacar que os resultados obtidos com os experimentos realizados com a cabeça robótica abordam diversas questões de pesquisa da área da robótica sociável. O sistema de visão desenvolvido durante este trabalho é capaz de simular as preferências visuais de uma criança entre os 6 e os 18 meses de idade. Este sistema, embora simples, apresenta características similares aos principais sistemas de visão propostos na literatura (Breazeal and Scassellati, 1999) (Breazeal, 2000) (Scassellati, 2001a) (Aggarwal and Cai, 1999) (Goodrich and Schultz, 2007). O sistema de voz desenvolvido é capaz reconhecer a linguagem natural falada no idioma Português Brasileiro, possibilitando que uma interação mais natural com o robô. Estes dois sistemas constituem uma interface multimodal capaz de fornecer os estímulos necessários e relevantes ao robô durante uma interação social, constituindo uma importante contribuição para a área. Como apontado anteriormente, os resultados obtidos pelos experimentos com o robô evidenciam que a arquitetura proposta é capaz de adquirir habilidades sociais básicas a

partir de comportamentos inatos existentes no repertório do robô e da interação com o ambiente. Os resultados evidenciam também que a arquitetura constitui uma contribuição para a área de pesquisa sobre o aprendizado da atenção compartilhada, por representar um modelo de aprendizado desta habilidade tida como um dos grandes desafios da robótica sociável (Kaplan and Hafner, 2004). Adicionalmente, deve-se ressaltar novamente que esta habilidade é considerada a base essencial para o desenvolvimento das habilidades sociais e cognitivas (Deák et al., 2001) (Smith and Ulvund, 2003) (Nagai et al., 2003a) (Kanda et al., 2004). Os experimentos realizados com o mecanismo de aprendizagem por tutela demonstraram como é possível transformar um problema de aprendizado de máquina em um problema de colaboração entre robôs e seres humanos, evidenciando como um diálogo colaborativo pode permitir que um robô aprenda conceitos sobre objetos ou eventos importantes do ambiente. Ainda, vale ressaltar que estes experimentos abordaram outra importante questão de pesquisa da área da robótica: o aprendizado de robôs sociáveis (Breazeal, 2002) (Breazeal, 2004) (Lockerd and Breazeal, 2004).

Durante os experimentos executados com a cabeça robótica interativa, ficaram evidentes algumas deficiências relativas à configuração da arquitetura, módulos de percepção e recursos computacionais.

A configuração manual e empírica do sistema motivacional tornou-se, como apontado na Seção 6.6, uma tarefa complexa e trabalhosa. O processo de configuração de todos os parâmetros da arquitetura foi executado segundo uma metodologia de tentativa e erro e demandou a execução de diversos testes de interação com robô por quatro dias, até o estabelecimento de uma configuração apropriada para a aplicação em questão. Entretanto, deve-se ressaltar que a configuração da base de conhecimento da arquitetura (regras de comportamento, regras de restrição, respostas, fatos e estímulos) foi muito facilitada pela forma de representação do conhecimento adotada.

O sistema de visão computacional apresentou diversas deficiências relativas à biblioteca responsável por detectar faces humanas e estimar a pose da cabeça (biblioteca *Watson*). Verificou-se, após a realização de diversos experimentos preliminares com o sistema completo, que o modelo inicial da face, formado pela *Watson* durante a sua inicialização, se degradava quando o robô perdia o contato com o ser humano, quando este direcionava seu olhar a outro lugar do ambiente para procurar uma fruta ou algum objeto. Este problema foi solucionado com a retirada da câmera responsável pela aquisição da imagem para a *Watson* e pelo posicionamento desta câmera de maneira a manter o foco permanente no ser humano. Desta forma, a câmera da *Watson* foi posicionada atrás do robô, em uma plataforma que permitia que esta capturasse as imagens da face do ser humano durante a interação com o robô.

Entretanto, esta solução pode implicar em problemas antropomórficos, uma vez que é mais natural para um ser humano que os “olhos” do robô estejam posicionados na face do mesmo. Portanto, vale apontar aqui a importância de uma pesquisa futura com o objetivo de se aprimorar a técnica empregada para a detecção da face e estimativa da pose.

Alguns problemas menores podem ser apontados sobre o sistema de visão, relativos ao módulo de detecção e reconhecimento de objetos. Na versão desenvolvida durante esta pesquisa, este módulo emprega apenas a detecção das cores dos objetos, tornando o mecanismo pouco robusto quanto à variação da luminosidade do ambiente. Portanto, vale apontar aqui a importância de uma pesquisa futura com o objetivo de se incorporar novas técnicas para a detecção de objetos, como a detecção de forma e textura.

Outra deficiência que pode ser apontada é relativa aos recursos computacionais. Todo os sistemas de controle, incluindo o sistema de visão, voz, arquitetura de controle e sistema motor, foi utilizado em um único computador: um notebook com um processador AMD de 1.80 Ghz, com 1.00 Gb de memória RAM, com o sistema operacional *Windows XP Pro*. Esta escolha foi motivada pelo fato de se poder transportar o sistema juntamente com o robô para lugares diferentes de maneira prática. Entretanto, esta configuração apresentou diversos problemas durante os experimentos, uma vez que o *Windows* apresentou dificuldades em tratar as diversas tarefas simultâneas em execução no computador, ocasionando muitas falhas de reconhecimento de voz e degradações do modelo da face do ser humano, além de alguns travamentos no sistema motor do robô. Embora esta deficiência de recursos não tenha impedido a realização dos experimentos com sucesso, pode-se ressaltar aqui a necessidade de se empregar ao menos três nós de processamento independentes para possibilitar a realização de interações sociais complexas com o robô.

Vale novamente ressaltar que este trabalho constitui a base para o desenvolvimento de uma ferramenta para construção de robôs sociáveis. A arquitetura, estruturas, métodos e módulos desenvolvidos no presente trabalho, permitem interações sociais limitadas em um ambiente social real e controlado. Portanto, é necessário o desenvolvimento e a integração de novos módulos sociais que permitam a aplicação da arquitetura e a construção de um robô sociável capaz de interagir de uma maneira mais natural em um ambiente real não estruturado. Trabalhos futuros podem incluir o desenvolvimento de novos mecanismos sociais que introduzam novas habilidades ao robô, como apresentado na próxima Seção.

Finalmente, vale ressaltar novamente que até o momento, segundo o levantamento bibliográfico feito pelo pesquisador e pelo grupo de pesquisa, não

existem trabalhos similares dentro da robótica sociável, sendo desenvolvidos no Brasil.

Principais Contribuições

Embora existam, na literatura, diversos trabalhos relacionados à robótica sociável, uma vez que esta tem se mostrado uma área de pesquisa muito ativa, pode-se considerar a pesquisa bibliográfica, conduzida durante este trabalho, como uma contribuição desta Tese, uma vez que procurou-se reunir os principais trabalhos relacionados à esta área de pesquisa. Adicionalmente, pode-se citar as seguintes contribuições resultantes deste trabalho:

- O desenvolvimento e a exploração de uma arquitetura robótica (Policastro et al., 2006) (Policastro et al., 2007a) (Policastro et al., 2007b) (Policastro et al., 2008b) que formará uma ferramenta para o projeto de robôs sociais. A arquitetura proposta apresenta os principais componentes encontrados nas mais importantes arquiteturas robóticas de robôs sociais encontradas na literatura, estando em conformidade com os principais resultados encontrados na área de pesquisa;
- O desenvolvimento e a exploração de algoritmos e técnicas inspirados nos processos evidenciados na Análise do Comportamento (Policastro et al., 2006) (Policastro et al., 2007a) (Policastro et al., 2007b). A arquitetura proposta é capaz de simular os principais processos básicos de aprendizado evidenciado na Análise do Comportamento, constituindo um importante resultado para as áreas da Psicologia e da Robótica;
- O desenvolvimento e a exploração de algoritmos baseados na técnica de aprendizado por reforço e a exploração novas alternativas para a representação do conhecimento (Policastro et al., 2006) (Policastro et al., 2007a) (Policastro et al., 2007b) (Policastro et al., 2008c). Embora o algoritmo desenvolvido não seja inteiramente inédito, este apresenta mecanismos e estruturas, inspiradas na Análise do Comportamento, que permitem o aprendizado pela consequência (aprendizado por contingência). Adicionalmente, o algoritmo de aprendizagem apresenta boa convergência, como pode-se verificar na discussão provida nesta Tese. Ainda, os mecanismos e estruturas de representação do conhecimento atendem a importantes questões sobre a representação do conhecimento, como a generalização incremental do conhecimento adquirido, por meio do emprego de uma rede neural auto-organizável. Como apontado anteriormente, estes algoritmos constituem um modelo computacional inédito do

aprendizado da atenção visual compartilhada e são uma importante contribuição para a área de pesquisa da robótica sociável, uma vez que esta habilidade é apontada como fundamental e um grande desafio desta área de pesquisa.

- O desenvolvimento de um sistema de visão e de voz e de um sistema motivacional para robôs sociais (Policastro et al., 2008a). Embora os sistemas de visão e voz e motivacional desenvolvidos neste trabalho sejam relativamente simples, em relação a outros sistemas existentes na literatura, os resultados obtidos com os experimentos empregando estes sistemas demonstram resultados promissores. Todos estes sistemas, mecanismos e estruturas desenvolvidos foram a base para o desenvolvimento de uma ferramenta para a construção de robôs sociáveis;
- O desenvolvimento de um simulador de interações sociais (Policastro et al., 2007b) (Policastro et al., 2008c) capaz de simular a interação entre um ser humano e um robô, no contexto do aprendizado da atenção compartilhada. Este simulador possibilitou a validação das estruturas e métodos da arquitetura robótica proposta nesta pesquisa, antes da utilização desta arquitetura de controle nos experimentos com o robô real. Desta forma, é possível validar o funcionamento de novos algoritmos antes de integrá-lo ao sistema de controle do robô, possibilitando o ajuste e o refino dos algoritmos de forma rápida e controlada. Este simulador integra o conjunto de ferramenta para a construção de robôs sociáveis;
- O desenvolvimento de um mecanismo de aprendizado por tutela (Policastro et al., 2008a). O mecanismo de aprendizagem proposto neste trabalho de pesquisa é capaz de associar estímulos visuais e auditivos para simular a aprendizagem de conceitos sobre objeto do real mundo por meio da tutela de um ser humano. Os resultados obtidos com os experimentos estão de acordo com a literatura da robótica sociável, especificamente sobre o desenvolvimento de abordagens de aprendizagem por meio de interações sociais.

Trabalhos Futuros

Baseado nas discussões apresentadas e na literatura existente, pode-se apontar alguns trabalhos futuros para abordar diversas questões de pesquisa da área da robótica sociável:

- *Aprendizado de robôs sociais.* Embora o presente trabalho de pesquisa aborde tal questão, ainda existem grandes desafios a serem superados

para possibilitar que um robô sociável possa ser ensinado da mesma maneira que um ser humano (ou a mais similar possível). Neste contexto, pode-se destacar como trabalho futuro a pesquisa e o desenvolvimento de métodos que possibilitem o aprendizado por imitação ou o desenvolvimento de métodos mais sofisticados que possibilitem o aprendizado por tutelação.

- *Detecção e Rastreamento de faces.* Com base nas deficiências apresentadas nas discussões sobre os experimentos com o robô, pode-se descartar que um grande desafio nesta área é o desenvolvimento de uma abordagem de rastreamento de faces humanas e da pose da cabeça que apresente robustez em diversas condições de iluminação e posição inicial de uma face detectada, suportando longas interações em ambientes não estruturados e dinâmicos.
- *Detecção de objetos.* Outro trabalho futuro que pode ser destacado, em relação a sistemas de percepção, envolve o aperfeiçoamento do atual módulo de detecção de objetos para incorporar forma e textura, tornando esta abordagem mais robusta à variação da luminosidade do ambiente.
- *Motivação.* Embora o sistema motivacional, desenvolvido durante este trabalho, tenha servido ao propósito de conferir pró-atividade ao robô sociável, a incorporação de emoções (como explorado em diversos trabalhos recentes da literatura), pode conferir uma maior robustez ao mesmo, além de possibilitar o desenvolvimento de robôs sociáveis capazes de realizar interações mais complexas e longas.
- *Interação social.* Se realmente o objetivo desta linha de pesquisa, iniciada no Instituto de Ciências Matemáticas e de Computação - ICMC-USP, é o desenvolvimento de um robô sociável, então pode-se apontar como um importante trabalho futuro o desenvolvimento de um módulo de interações de longo prazo, integrando motivação, emoções, visão e voz.

Espera-se que novos trabalhos possam se utilizar dos problemas de pesquisa citados anteriormente, bem como que o presente trabalho contribua, efetivamente, para o desenvolvimento da área da robótica sociável em interação com outras áreas do conhecimento, como a Psicologia.

Referências

- Aggarwal, J. and Cai, Q. (1999). Human motion analysis: A review. *Computer Vision and Image Understanding*, 73(3):428–440.
- Albus, J., Lumia, R., Fiala, J., and Wavering, A. (1989). NASREM: The nasa/nbs standard reference model for telerobot control system architecture. In *20th International Symposium on Industrial Robots, Tokyo, Japan*.
- Antonitis, J. (1951). Response variability in the rat during conditioning, extinction, and reconditioning. *Journal of Experimental Psychology*, 42:273–281.
- Arkin, C. (1989). Motor schema-based mobile robot navigation. *Journal of Robotics Research*, 8(4):92–112.
- Arkin, R. (1998). *Behavior-Based Robotics*. MIT Press, Cambridge.
- Arkin, R. and Balch, T. (1997). Aura: Principles and practice in review. *Journal of Experimental and Theoretical Artificial Intelligence(JETAI)*, 9(2/3):175–188.
- Arkin, R., Fujita, M., Takagi, T., and Hasegawa, R. (2001). Ethological modeling and architecture for an entertainment robot. In *IEEE International Conference on Robotics and Automation, Seoul, Korea*.
- Arkin, R., Fujita, M., Takagi, T., and Hasekawa, R. (2003). An ethological and emotional basis for human-robot interaction. *Robotics and Autonomous Systems*, 42:191–201.
- Björne, P. and Balkenius, C. (2005). A model of attentional impairments in autism: first steps toward a computational theory. *Cognitive Systems Research*, 6(3):193–204.
- Blockeel, H., Raedt, L., and Ramon, J. (1998). Top-down induction of clustering trees. In *15th International Conference on Machine Learning*. Morgan Kaufmann.

- Bosa, C. (2002). Atenção compartilhada e identificação precoce do autismo. *Psicologia: Reflexão e Crítica*, 15(1):77--88.
- Breazeal, C. (2000). *Sociable Machines: Expressive Social Interaction Between Human and Robots*. PhD thesis, MIT Department of Electrical Engineering and Computer Science, Cambridge, MA.
- Breazeal, C. (2001). Socially intelligent robots: Research, development, and applications. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 2121--2126.
- Breazeal, C. (2002). *Designing Sociable Robots*. MIT Press.
- Breazeal, C. (2003a). Emotion and sociable humanoid robots. *Human-Computer Studies*, 59(1):119-155.
- Breazeal, C. (2003b). Toward sociable robots. *Robotics and Autonomous Systems*, 42(3-4):167--175.
- Breazeal, C. (2004). Social interactions in hri: the robot view. *IEEE Transactions on Man and Cybernetics , Part C: Applications and Reviews*, 34(2):181--186.
- Breazeal, C., Berlin, M., Brooks, A., Gray, J., and Thomaz, A. (2006). Using perspective taking to learn from ambiguous demonstrations. *Robotics and Autonomous Systems*, 54:385--393.
- Breazeal, C., Buchsbaum, D., Gray, J., Gatenby, D., and Blumberg, B. (2005). Learning from and about the others: Towards using imitation to bootstrap the social understanding of others by robots. *Artificial Life*, 11(1-2):31--62.
- Breazeal, C., Edsinger, A., Fitzpatrick, P., and Scassellati, B. (2001). Active vision for sociable robots. *IEEE Trans. Systems, Man, and Cybernetics*, 31(5):443--453.
- Breazeal, C. and Scassellati, B. (1999). A context-dependent attention system for a social robot. In *International Joint Conference on Artificial Intelligence*, pages 1146--1153.
- Breazeal, C. and Scassellati, B. (2000). Infant-like social interactions between a robot and a human caretaker. *Adaptive Behavior*, 8(1):49--74.
- Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2:14--23.
- Brooks, R. (1990). Challenges for complete creature architecture. In *1st International Conference on Simulation of Adaptive Behavior*, pages 434--443. MIT Press, Cambridge.

- Brooks, R. (1991). Intelligence without reason. In *International Joint Conference on Artificial Intelligence*, pages 569–595.
- Bruce, A., Nourbakhsh, I., and Simmons, R. (2001). The role of expressiveness and attention in human-robot interaction. In *AAAI Fall Symposium Emotional and Intelligent II: The Tangled Knot of Society of Cognition*.
- Burt, P. and Adelson, E. (1983). The laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31:532–540.
- Carbonell, J., Knoblock, C., and Minton, S. (1989). PRODIGY: An integrated architecture for planning and learning. *Carnegie Mellon Technical Report, CMU-CS-89-189*.
- Carlson, E. and Triesch, J. (2003). A computational model of the emergence of gaze following. In *8th Neural Computation Workshop (NCPW8)*.
- Carpenter, G. and Grossberg, S. (1987). ART 2 : Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 26:23:4919–4930.
- Carver, N. and Lesser, V. (1994). The evolution of blackboard control architectures. *Expert Systems with Applications—Special Issue on the Blackboard Paradigm and Its Applications*, 7(1):1–30.
- Catania, A. (1999). *Aprendizagem: Comportamento, Linguagem e Cognição*. Artmed.
- Chapman, D. and Kaelbling, L. (1991). Input generalization in delayed reinforcement learning: An algorithm and performance comparisons. In *Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, San Mateo, Ca, pages 726–731. Morgan Kaufmann.
- Collins, H. (1990). *Artificial Experts: Social Knowledge and Intelligent Machines*. MIT Press.
- Dautenhahn, K. (1995). Getting to know each other - artificial social intelligence for autonomous robots. *Robotics and Autonomous Systems*, 16:333–356.
- Dautenhahn, K. (1997). I could be you - the phenomenological dimension of social understanding. *The Cybernetics and Systems Journal*, 28(5).
- Dautenhahn, K. (1998). The art of designing socially intelligent agents - science, fiction and the human in the loop. *Applied Artificial Intelligence Journal*, 12(7–8):573–617.
- Dautenhahn, K. (2000). Socially intelligent agents and the primate social brain - toward a science of social minds. In *AAAI Fall Symposium on Society of Intelligence Agents*.

- Dautenhahn, K. and Billard, A. (1999). Bringing up robots or—the psychology of socially intelligent robots: From theory to implementation. In *Autonomous Agents*.
- Dautenhahn, K., Ogden, B., and Quick, T. (2002). From embodied to socially embedded agents-implications for interaction-aware robots. *Cognitive Systems Research*, 3(3).
- Deák, G., Bartlett, M., and Jebara, T. (2007). New trends in cognitive science: Integrative approaches to learning and development. *Neurocomputing*, 70:21392147.
- Deák, G., Fasel, I., and Movellan, J. (2001). The emergence of shared attention: Using robots to test developmental theories. In Balkenius, C., Zlatev, J., Kozima, H., Dautenhahn, K., and Breazeal, C., editors, *First International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*, pages 95–104. Lund: Lund University.
- Deák, G. and Triesch, J. (2005). Origins of shared attention in human infants. In Fuzita, K. and Itakura, S., editors, *Diversity of Cognition (In Press)*, pages 67–74. Kyoto University Press.
- Driessens, K. (2004). *Relational reinforcement learning*. PhD thesis, Department of Computer Science, K.U.Leuven, Leuven, Belgium.
- Dube, W., McDonald, R., Mansfield, R., Holcomb, W., and Ahearn, W. (2004). Toward a behavioral analysis of joint attention. *The Behavior Analyst*, 27(2):197–207.
- Duffy, B. (2003). Anthropomorphism and the social robot. *Robotics and Autonomous Systems*, 42:177–190.
- Duffy, B., Dragone, M., and O'Hare, G. (2005). Social robot architecture: A framework for explicit social interaction. In *Android Science: Towards Social Mechanisms, CogSci 2005 Workshop*, Stresa, Italy.
- Dzeroski, S. (2003). Relational reinforcement learning for agents in worlds with objects. *Lecture Notes in Computer Science*, 2636:306–322.
- Dzeroski, S., Raedt, L., and Driessens, K. (2001). Relational reinforcement learning. *Machine Learning*, 43:7–52.
- Ekman, P. (1999). *Basic emotions*. Wiley.
- Fasel, I., Deak, G., Triesch, J., and Movellan, J. (2002). Combining embodied models and empirical research for understanding the development of shared attention. In *2nd International Conference on Development and Learning*, pages 21–27.

- Freeman, W. and Adelson, E. (1991). The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:891–906.
- Gadano, S. and Hallan, J. (2002). Robot learning driven by emotions. *Adaptive Behavior*, 9(1):42–64.
- Gat, E. (1992). Integrating planning and reacting in a heterogeneous asynchronous architecture for controlling real-world mobile robots. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*.
- Geen, R. (1994). *Human motivation: A psychological approach*. Wadsworth Publishing.
- Gockley, R., Bruce, A., Forlizzi, J., Michalowski, M., Mundell, A., Rosenthal, S., Sellner, B., Simmons, R., Snipes, K., Schultz, A., and Wang, J. (2005). Designing robots for long-term social interaction. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS '05)*. IEEE.
- Gockley, R., Forlizzi, J., and Simmons, R. (2007). Natural person-following behavior for social robots. In *Proceeding of the ACM/IEEE international conference on Human-robot interaction, Arlington, Virginia, USA*, pages 17–24. ACM Press.
- Gockley, R., Simmons, R., and Forlizzi, J. (2006). Modeling affect in socially interactive robots. In *15th IEEE International Symposium on Robot and Human Interactive Communication*, pages 558–563.
- Gold, K. and Scassellati, B. (2007). A bayesian robot that distinguishes “self” from “other”. In *To appear in the Proceedings of the 29th Annual Meeting of the Cognitive Science Society (CogSci'07)*.
- Goldberg, D. (1989). *Genetic Algorithm in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, Inc., Massachusetts.
- Goodrich, M. and Schultz, A. (2007). Human-robot interaction: A survey. *Foundations and Trends in Human-Computer Interaction*, 1(3):203–275.
- Haykin, S. (1999). *Neural Networks - A Comprehensive Foundation*. Prentice Hall.
- Ishiguro, H., Kanda, T., Kimoto, K., and Ishida, T. (1999). A robot architecture based on situated modules. In *International Conference on Intelligent Robots and Systems, 1999*, pages 1617–1624, Kyongju, South Korea.
- Itti, L., Koch, C., and Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259.

- Kaelbling, L., Littman, M., and Moore, A. (1996). Reinforcement learning: A survey. *Journal of AI Research*, 4:237--285.
- Kanda, T., Sata, R., Saiwaki, N., and Ishiguro, H. (2004). Friendly social robot that understands human's friendly relationships. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2004)*, pages 2215--2222.
- Kaplan, F. (2001). Taming robots with clicker training: A solution for teaching complex behaviors. In *the European Workshop on Learning Robots*.
- Kaplan, F. and Hafner, V. (2004). The challenges of joint attention. In Berthouze, L., Kozima, H., Prince, C., Sandini, G., Stojanov, G., Metta, G., and Balkenius, C., editors, *4th International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic System*, pages 67--74.
- Kobayashi, H., Hara, F., and Tange, A. (1994). A basic study on dynamic control of facial expressions for face robot. In *International Workshop on Robots and Human Communication*.
- Laird, J., Newell, A., and Rosenbloom, P. (1987). SOAR: an architecture for general intelligence. *Artificial Intelligence*, 33(1):1--64.
- Lau, B. and Triesch, J. (2004). Learning gaze following in space: a computational model. In *3rd International Conference for Development and Learning, ICDL'04*.
- Lauria, S., Bugmann, G. and Kyriacou, T., and Klein, E. (2002). Mobile robot programming using natural language. *Robotics and Autonomous Systems*, 38(3):171--181.
- Lawrence, R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of IEEE*, 77(2):257--286.
- Lehman, J.F. and Laird, J. and Rosenbloom, P. (2005). A gentle introduction to soar, an architecture for human cognition. *Cognitive Science*, 5.
- Lindblom, J. and Ziemke, T. (2003). Social situatedness of natural and artificial intelligence: vygotsky and beyond. *Adaptative Behavior*, 11(2):79--96.
- Lockerd, A. and Breazeal, C. (2004). Tutelage and socially guided robot learning. In *International Conference on Intelligent Robots and Systems (IROS'2004)*, pages 3475--3480.
- Lorenz, K. (1998). The foundations of ethology. In *Workshop on Robots as Partners: An Exploration of Social Robots*. Springer.

- Lti-Lib (2003). Lti image processing library - developer's guide. Technical report, Lehrstuhl fuer Technische Informatik - Aachen University of Technology.
- Lungarella, M. and Metta, G. (2003). Beyond gazing, pointing, and reaching: a survey of developmental robotics. In *3rd International Workshop on Epigenetic Robotics*, pages 81--89.
- Marom, Y. and Hayes, G. (2001). Attention and social situatedness for skill acquisition. Technical Report EDI-INF-RR-0069, University of Edinburgh.
- Mataric, M. (1999). Behavior-based robotics. In Wilson, R. and Keil, F., editors, *MIT Encyclopedia of Cognitive Sciences*, pages 74--77. MIT Press.
- Michaud, F. and Caron, S. (2000). Roball-an autonomous toy-rolling robot. In *Workshop on Interactive Robot Entertainment*.
- Mitchell, T. (1997a). Evaluating hypotheses. In *Machine Learning. Capítulo 5*. McGraw-Hill.
- Mitchell, T. (1997b). *Machine Learning*. McGraw-Hill.
- Morency, L.-P. (2007). User guide - head tracking and gesture recognition library. Technical report, MIT Department of Electrical Engineering and Computer Science.
- Morency, L.-P. and Darrell, T. (2002). Stereo tracking using icp and normal flow constraint. In *International Conference on Pattern Recognition*.
- Morency, L.-P., Sundberg, P., and Darrel, T. (2003). Pose estimation using 3d view-based eigenspaces. In *Analysis and Modeling of Faces and Gestures (AMFG'03)*, pages 45--52. IEEE.
- Moses, L. (1986). Comparison of averages from two samples and some related problems. In *Think and Explain with Statistics. Capítulo 6*. Addison - Wesley.
- Nagai, Y., Hosoda, A., and Asada, M. (2003a). A constructive model for the development of joint attention. *Connection Science*, 15(4):211--229.
- Nagai, Y., Hosoda, K., and Asada, M. (2003b). How does an infant acquire the ability of joint attention?: A constructive approach. In *3rd International Workshop on Epigenetic Robotics*, pages 91--98.
- Newell, A. and Simon, H. (1976). Computer science as empirical inquiry: Symbols and search. *Commun. Assoc. Comput. Machinery*, 19:111--126.
- Nuance (2001). Introduction to the nuance system. Technical report, Nuance Communications, Inc.

- Otterlo, V. (2005). A survey of reinforcement learning in relational domains. Technical Report TR-CTIT-0531, University of Twente, The Netherlands.
- Oza, N. (1999). A survey of robot architectures. cite-seer.ist.psu.edu/205448.html.
- Peng, J. and Williams, R. (1996). Incremental multi-step q-learning. *Machine Learning*, 22:283--290.
- Policastro, C., Carvalho, A., and Delbem, A. (2003). Hybrid approaches for case retrieval and adaptation. In *26th Annual German Conference on AI, KI 2003, Hamburg, Germany*, volume 2821 of *Lecture Notes in Computer Science*, pages 297--311. Springer.
- Policastro, C., Pais, G., Munhoz, V., Romero, R., Zuliani, G., and Pizzolato, E. (2008a). Applying learning by tutelage and multimodal interface to sociable robots. In *IEEE Conference Conference on Machine Learning and Applications (ICMLA' 08)*. To be Published. IEEE.
- Policastro, C., Pais, G., Munhoz, V., Romero, R., Zuliani, G., and Pizzolato, E. (2008b). Learning of shared attention in sociable robotics. *Em submissão ao Journal of Algorithms - Thematic Issue on Algorithmic Reinforcement Learning*.
- Policastro, C., Zuliani, G., da Silva, R., Munhoz, V., and Romero, R. (2008c). Hybrid knowledge representation applied to the learning of the shared attention. In *IEEE World Congress on Computational Intelligence (WCCI'08), Hong Kong*. To be published. IEEE.
- Policastro, C., Zuliani, G., and Romero, R. (2006). Arquitetura robótica inspirada na análise do comportamento. In *1st Workshop on Computational Intelligence (WCI'06) in the Proceedings of International Joint Conference, 10th Ibero-American Artificial Intelligence Conference, 18th Brazilian Artificial Intelligence Symposium, 9th Brazilian Neural Networks Symposium, IBERAMIA-SBIA-SBRN, Ribeirão Preto, Brazil*.
- Policastro, C., Zuliani, G., and Romero, R. (2007a). Robotic architecture inspired on behavior analysis. In *IEEE International Joint Conference on Neural Network (IJCNN'07), Orlando, Florida, USA*, pages 1482--1487. IEEE.
- Policastro, C., Zuliani, G., and Romero, R. (2007b). Robotic architecture inspired on behavior analysis. *Em submissão ao Adaptive Behavior Journal*.
- Rabiner, L. and Juang, B. (1993). Fundamentals of speech recognition. In *Englewood Cliffs*. Prentice-Hall.
- Restivo, S. (2001). Bringing up and booting up: Social theory and the emergence of socially intelligent robots. In *Proceedings of the 2001 Systems*,

Man, and Cybernetics Conference, Tucson AZ.

- Robins, B., Dickerson, P., Stribling, P., and Dautenhahn, K. (2004). Robot-mediated joint attention in children with autism: A case study in robot-human interaction. *Interaction Studies*, 5(2):161–198.
- Rodrigues, F. and Gomes, H. (2002). Applying a visual attention mechanism to the problem of traffic sign recognition. In *SIBGRAPI*, page 415.
- Rosenblatt, J. (1995). Damn: A distributed architecture for mobile navigation. In *AAAI Spring Symposium on Lessons Learned from Implemented Software Architectures for Physical Agents*.
- Russell, S. and Norvig, P. (2003). *Artificial Intelligence - A Modern Approach*. Prentice-Hall.
- Sabanovic, S., Michalowski, M., and Simmons, R. (2006). Robots in the wild: observing human-robot social interaction outside the lab. In *Proceedings of the 9th International Workshop on Advanced Motion Control (AMC 2006)*, pages 596–601. IEEE.
- Sabanovic, S., Piotr Michalowski, M., and Caporael, L. (2007). Making friends: Building social robots through interdisciplinary collaboration. In *Multidisciplinary Collaboration for Socially Assistive Robotics: Papers from the 2007 AAAI Spring Symposium, Technical Report SS-07-07*, pages 71–77. AAAI.
- Salichs, M., Barber, R., Khamis, A., Malfaz, M., Gorostiza, J., Pacheco, R., Rivas, R., Corrales, A., and Delgado, E. (2006). Maggie: A robotic platform for human-robot social interaction. In *IEEE International Conference on Robotics, Automation and Mechatronics (RAM 2006)*. Bangkok, Thailand.
- Scassellati, B. (2000). Investigating models of social development using a humanoid robot. In Webb, B. and Consi, T., editors, *Biorobotics*, MIT Press.
- Scassellati, B. (2001a). *Foundations for a theory of mind for a humanoid robot*. PhD thesis, MIT Department of Electrical Engineering and Computer Science, Cambridge, MA.
- Scassellati, B. (2001b). Using social robots to study abnormal social development. In Berthouze, L., Kaplan, F., Kozima, H., Yano, H., Konczak, J., Metta, G., Nadel, J., Sandini, G., Stojanov, G., and Balkenius, C., editors, *Fifth International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems, Nara, Japan*, pages 11–14.
- Scheeff, M. (2000). Experiences with sparky: A social robot. In *Workshop on Interactive Robot Entertainment*.

- Serio, T., Andery, M., Gioia, P., and Micheleto, N. (2004). *Controle de estímulos e comportamento operante: uma (nova) introdução*. EDUC - Editora da PUC - SP.
- Shaw, M. and Garlan, D. (1996). *Software Architecture, Perspectives on an Emerging Discipline*. Prentice Hall.
- Sidner, C., Kidd, D., Lee, C., and Lesh, N. (2004). Where to look: A study of human-robot engagement. In *9th international conference on Intelligent user interface*, pages 78--84.
- Sidner, C., Lee, C., Kidd, C., Lesh, N., and Rich, C. (2005). Explorations in engagement for humans and robots. *Artificial Intelligence*, 166(1-2):181--186.
- Skinner, B. (1953). *Science and human behavior*. New York: The Macmillan Company.
- Skinner, B. (1974). *About Behaviorism*. Penguin Books.
- Smith, L. and Ulvund, S. (2003). The role of joint attention in later development among preterm children: Linkages between early and middle childhood. *Social Development*, 12(2).
- Staddon, J. (1983). *Adaptive Behavior and Learning*. Cambridge Press.
- Sutton, R. and Barto, A. (1998). *Reinforcement Learning: An Introduction*. MIT Press.
- Thorndike, E. (1898). A tutorial on hidden markov models and selected applications in speech recognition. *Psychological Review, Monograph Supplements*, 2(4).
- Triesch, J., Teuscher, C., Deák, G., and Carlson, E. (2006). Gaze following: why (not) learn it? *Developmental Science*, 2(9):125--157.
- Waldherr, S., Romero, R., and Thrun, S. (2000). A gesture-based interface for human-robot interaction. *Autonomous Robots*, 9.
- Watkins, C. (1989). *Models of delayed reinforcement learning*. PhD thesis, Psychology Department, Cambridge University.
- Webb, B. (2000). What does robotics offer animal behaviour? *Animal Behaviour*, 60:545--558.
- Whalen, C. and Schreibman, L. (2003). Joint attention training for children with autism using behavior modification procedures. *Journal of Child Psychology and Psychiatry*, 44(3):456--468.
- Wu, Y. and Huang, T. (1999). Vision-based gesture recognition: A review. In Springer, editor, *Gesture-Based Communications in HCI. LNCS 1739*.

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)