

**SILVANA VALDEMARA APARECIDA MICHELOTTO
SAVARIS**

**SISTEMA ESPECIALISTA PARA PRIMEIROS
SOCORROS PARA CÃES**

Florianópolis - SC
2002

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

Silvana Valdemara Aparecida Michelotto Savaris

**SISTEMA ESPECIALISTA PARA PRIMEIROS
SOCORROS PARA CÃES**

Dissertação Submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação

Professor Jorge Muniz Barreto, D.Sc.A.

Florianópolis, Julho de 2002

SISTEMA ESPECIALISTA PARA PRIMEIROS SOCORROS PARA CÃES

Silvana Valdemara Aparecida Michelotto Savaris

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, Área de Concentração Sistemas de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Professor Fernando Álvaro Ostuni Gauthier, Dr.
Coordenador do Curso

Banca Examinadora

Professor Jorge Muniz Barreto, D.Sc.A.
Orientador

Professora Maria Augusta Soares Machado, Dra.

Professor Mauro Roisenberg, Dr.

Professor Raul César Baptista Machado, Dr.

“Não é o desafio com que nos deparamos que determina quem somos e o que estamos nos tornando, mas a maneira com que respondemos ao desafio. Somos combatentes, idealistas, mas plenamente conscientes. Porque termos consciência, não nos obriga a ter teoria sobre as coisas: só nos obriga a sermos conscientes. Problemas para vencer, liberdade para provar. E, enquanto acreditamos no nosso sonho, nada é por acaso.”

*A meus pais, meu irmão e meu esposo,
por toda força, carinho e apoio
em todos os momentos,
suportando minha ausência em
momentos importantes.
Obrigado pela compreensão.*

Primeiramente, agradeço a Deus e a Nossa Senhora Aparecida, pela luz fortalecente; a energia positiva para enfrentar todos os obstáculos desta árdua caminhada e ao final dizer com orgulho Eu Venci, Valeu a Pena Lutar.

A meus pais, Juscelina e Santo, e ao meu irmão, Rodrigo, por proporcionarem estudo, apoio à escolha profissional e pela compreensão, carinho, amor e incentivo dispendidos durante toda minha vida para chegar aonde cheguei. Tudo eu devo a vocês. Vocês são minha luz, minha vida. Amo Vocês.

A meu esposo, Moacir, pela paciência. Você soube aceitar como ninguém minha ausência, incentivando-me porque acreditava no meu potencial e sabia que eu iria vencer, passando horas nos finais de semana ao meu lado, enquanto escrevia minha dissertação. Seu apoio e carinho foram os fatores primordiais neste período de sacrifícios a fim de eu realizar um ideal esperado. Amo Você.

A minha sogra, Dona Rosalina, que sempre me apoiou e incentivou, e hoje, in memoriam, continua me guiando com sua luz, mostrando-me o caminho do saber e dando-me forças que concluir com êxito o que ela pode ver Eu iniciar.

A meu orientador, pela orientação, acompanhamento, apoio e dedicação, indicando sempre os melhores caminhos para que pudesse chegar até o fim com sucesso.

A todas minhas amigas e amigos, pela amizade, companheirismo.

Aos médicos veterinários, pelas informações passadas, necessárias para o desenvolvimento do sistema.

A Universidade Federal de Santa Catarina (UFSC) e a Universidade do Oeste de Santa Catarina (UNOESC), por proporcionarem o curso de Mestrado, incentivando os alunos a iniciarem, contribuindo no aperfeiçoamento profissional, e, objetivando formar novos profissionais aptos para ingressarem no mercado de trabalho.

SUMÁRIO

I - INTRODUÇÃO.....	1
1.1 - MOTIVAÇÃO.....	1
1.2 - OBJETIVOS.....	2
1.2.1 - OBJETIVO GERAL.....	2
1.2.2 - OBJETIVOS ESPECÍFICOS.....	2
II - SISTEMAS ESPECIALISTAS.....	4
2.1 - HISTÓRICO.....	4
2.2 - ALGUNS SE DESENVOLVIDOS NA UNIVERSIDADE FEDERAL DE SANTA CATARINA (UFSC)....	5
2.3 - PUBLICAÇÕES DE SE DESENVOLVIDOS NA UNIVERSIDADE FEDERAL DE SANTA CATARINA (UFSC).....	8
2.4 - CONCEITOS.....	11
2.5 - QUANDO OS SISTEMAS ESPECIALISTAS SÃO UTILIZADOS.....	14
2.6 - A EFICÁCIA DOS SISTEMAS ESPECIALISTAS.....	15
2.7 - PRINCIPAIS BENEFÍCIOS DA UTILIZAÇÃO DOS SISTEMAS ESPECIALISTAS.....	15
2.8 - PROBLEMAS ENFRENTADOS PELOS SISTEMAS ESPECIALISTAS ATUAIS.....	17
2.9 - CLASSIFICAÇÃO DE SISTEMAS ESPECIALISTAS.....	18
2.10 - FERRAMENTAS PARA IMPLEMENTAÇÃO DE SE.....	20
2.10.1 - SHELL DE SISTEMAS ESPECIALISTAS.....	20
2.11 - CICLO DE VIDA DE SISTEMAS ESPECIALISTAS.....	22
2.11.1 - FASES DO DESENVOLVIMENTO DE UM SISTEMA ESPECIALISTA.....	22
2.11.1.1 - FASE DE IDENTIFICAÇÃO.....	22
2.11.1.1.1 - IDENTIFICAÇÃO DOS RECURSOS.....	22
2.11.1.1.2 - IDENTIFICAÇÃO DAS CARACTERÍSTICAS DO PROBLEMA.....	23
2.11.1.2 - FASE DA CONCEITUAÇÃO.....	23
2.11.1.3 - FASE DA FORMALIZAÇÃO.....	24
2.11.1.4 - FASE DA IMPLEMENTAÇÃO.....	24
2.11.1.5 - FASE DO TESTE E AVALIAÇÃO.....	24
2.11.1.6 - FASE DA REVISÃO.....	26
III - ARQUITETURA E SISTEMAS ESPECIALISTAS SIMBÓLICOS.....	27
3.1 - SISTEMAS ESPECIALISTAS SIMBÓLICOS (SES).....	27
3.2 - SISTEMAS ESPECIALISTAS CONEXIONISTAS (SEC).....	27
3.3 - UMA COMPARAÇÃO SIMPLIFICADA ENTRE A ESTRUTURA DE UM SES E DE UM SEC.....	28
3.4 - CONHECIMENTO.....	33
3.4.1 - COMPARAÇÃO ENTRE CONHECIMENTO ESPECIALIZADO HUMANO E ARTIFICIAL.....	34
3.4.2 - CONHECIMENTO E SUA REPRESENTAÇÃO.....	36
3.4.2.1 - PORQUE REPRESENTAR O CONHECIMENTO?.....	37
3.4.2.2 - CARACTERÍSTICAS DE BOAS REPRESENTAÇÕES DE CONHECIMENTO.....	37
3.4.3 - PRINCIPAIS FORMAS DE REPRESENTAÇÃO DO CONHECIMENTO.....	39
3.4.3.1 - REPRESENTAÇÃO DO CONHECIMENTO EM IAC.....	39
3.4.3.2 - REPRESENTAÇÃO DO CONHECIMENTO EM IAS.....	40
3.4.3.2.1 - SISTEMAS DE PRODUÇÃO.....	40
3.4.3.2.1.1 - VANTAGENS E DESVANTAGENS DAS REGRAS DE PRODUÇÃO.....	41
3.4.3.2.1.2 - DESVANTAGENS DOS SISTEMAS DE PRODUÇÃO.....	41
3.4.3.2.2 - REDES SEMÂNTICAS.....	42
3.4.3.2.3 - MOLDURAS (“FRAMES”).....	45
3.4.3.2.4 - ROTEIROS.....	47
3.4.3.2.5 - HERANÇA COMO REPRESENTAÇÃO DO CONHECIMENTO.....	48
3.4.3.2.6 - LÓGICA DAS PROPOSIÇÕES E DOS PREDICADOS.....	48
3.4.3.2.6.1 - LÓGICA DAS PROPOSIÇÕES.....	48
3.4.3.2.6.2 - LÓGICA DOS PREDICADOS.....	49
3.4.4 - OBSERVAÇÕES.....	51
3.4.4.1 - REPRESENTAÇÃO PROCEDIMENTAL.....	51
3.4.4.2 - DEPENDÊNCIA CONCEITUAL.....	54

3.5 - MECANISMOS DE RACIOCÍNIO.....	60
3.5.1 - RACIOCÍNIO COM REGRAS DE PRODUÇÃO.....	61
3.5.1.1 - EXEMPLO DE RACIOCÍNIO COM REGRAS DE PRODUÇÃO	63
3.5.1.2 - TÉCNICAS DE BUSCA	65
3.5.1.2.1 - GERAR E TESTAR.....	65
3.5.1.2.2 - RACIOCÍNIO PARA FRENTE VERSUS RACIOCÍNIO PARA TRÁS	65
3.5.1.2.3 - BUSCA EM LARGURA	66
3.5.1.2.4 - BUSCA EM PROFUNDIDADE.....	66
3.5.1.2.5 - SUBIDA DA MONTANHA	67
3.5.1.2.6 - ABDUÇÃO	67
3.5.2 - RACIOCÍNIO PROBABILÍSTICO.....	68
3.5.3 - RACIOCÍNIO QUALITATIVO	72
3.5.3.1 - ONTOLOGIAS DO RACIOCÍNIO QUALITATIVO	76
3.5.4 - RACIOCÍNIO NEBULOSO.....	78
3.5.4.1 - LÓGICA NEBULOSA E PROBABILIDADE.....	87
3.5.5 - RACIOCÍNIO PLAUSÍVEL.....	88
3.5.6 - RACIOCÍNIO NÃO-MONOTÔNICO.....	95
3.5.6.1 - INTRODUÇÃO AO RACIOCÍNIO NÃO-MONOTÔNICO	95
3.5.7 - RACIOCÍNIO ESTATÍSTICO E PROBABILÍSTICO	101
IV - SISTEMA ESPECIALISTA PARA PRIMEIROS SOCORROS PARA CÃES (SEPRISCA)	102
4.1 - INTRODUÇÃO	102
4.2 - ARQUITETURA DE UM SISTEMA ESPECIALISTA NO EXPERT SINTA.....	104
4.3 - GERENCIANDO BASES	104
4.3.1 - A JANELA “KNOWLEDGE-IN-A-BOX” (KIB).....	105
4.3.1.1 - CRIANDO VA RIÁVEIS.....	106
4.3.1.2 - DEFININDO O(S) OBJETIVO(S).....	107
4.3.1.3 - TRABALHANDO COM REGRAS.....	107
4.3.1.4 - DEFININDO INTERFACE COM O USUÁRIO.....	110
4.3.1.5 - INFORMAÇÕES ADICIONAIS SOBRE A BASE.....	112
4.3.1.6 - OPÇÕES.....	113
4.4 - CONSULTANDO O SISTEMA ESPECIALISTA.....	116
4.4.1 - CONCEITOS RÁPIDOS.....	117
4.4.2 - GUIA VISUAL DE OPERAÇÕES	117
4.4.3 - A CONSULTA	119
4.4.4 - AS JANELAS DE ACOMPANHAMENTO.....	120
4.4.5 - COMPREENDENDO OS RESULTADOS ATINGIDOS.....	122
4.5 - O DOMÍNIO DE CONHECIMENTO (SAÚDE DE CÃES).....	125
V - DISCUSSÃO E CONCLUSÕES.....	126
5.1 - ANÁLISE DOS RESULTADOS	126
5.2 - DISCUSSÃO.....	126
5.3 - CONCLUSÕES	127
VI - ANEXOS.....	129
6.1 - ANEXO 01 - UMA SUCINTA DESCRIÇÃO DA BASE DE CONHECIMENTOS DO SISTEMA SEPRISCA.....	129
6.2 - ANEXO 02 - UMA SUCINTA DEMONSTRAÇÃO DA EXECUÇÃO DO SISTEMA SEPRISCA.	132
VII - REFERÊNCIAS BIBLIOGRÁFICAS	135

LISTA DE FIGURAS

FIGURA 01 - ARQUITETURA DE UM SISTEMA ESPECIALISTA SIMBÓLICO (SES).....	28
FIGURA 02 - ARQUITETURA DE UM SISTEMA ESPECIALISTA CONEXIONISTA (SEC).....	32
FIGURA 03 - UMA REDE SEMÂNTICA.....	42
FIGURA 04 - REPRESENTAÇÃO LISP DE UMA REDE SEMÂNTICA	43
FIGURA 05 - UMA REDE SEMÂNTICA PARA UM PREDICADO DE N-ARGUMENTOS	44
FIGURA 06 - UMA REDE SEMÂNTICA REPRESENTANDO UMA FRASE.....	45
FIGURA 07 - MOLDURA PARCIAL DE UMA SALA DE ESTAR.....	46
FIGURA 08 - UMA REPRESENTAÇÃO PROCEDIMENTAL DE UMA FRASE DESCRITIVA	52
FIGURA 09 - UM EXEMPLO DE PROCEDIMENTO DE LIGAÇÃO	54
FIGURA 10 - AS DEPENDÊNCIAS DE CD.....	59
FIGURA 11 - UTILIZAÇÃO DE TEMPOS CONCEITUAIS.....	60
FIGURA 12 - ÁRVORE DE BUSCA	64
FIGURA 13 - GRAFOS DIRECIONADOS.....	71
FIGURA 14 - FÍSICA QUALITATIVA	72
FIGURA 15 - ESPAÇOS QUANTITATIVOS.....	74
FIGURA 16 - COEFICIENTES DE MUDANÇA	74
FIGURA 17 - OPERAÇÃO COM VARIÁVEIS QUALITATIVAS.....	74
FIGURA 18 - VALORES LIMITES ENTRE DUAS REGIÕES ADJACENTES	75
FIGURA 19 - TREZE RELACIONAMENTOS POSSÍVEIS ENTRE DOIS INTERVALOS DE TEMPO.....	76
FIGURA 20 - REPRESENTAÇÃO QUALITATIVA DA PRESSÃO SISTÓLICA	77
FIGURA 21 - FUNÇÕES DE PERTINÊNCIA PARA A PRESSÃO SISTÓLICA	78
FIGURA 22 - FUNÇÃO DE PERTINÊNCIA	79
FIGURA 23 - SISTEMA DE INFERÊNCIA NEBULOSA	80
FIGURA 24 - FORMAS DE REPRESENTAÇÃO DE CONJUNTOS NEBULOSOS	81
FIGURA 25 - POSIÇÃO DA CABINA DO CAMINHÃO EM RELAÇÃO A DOCA.....	82
FIGURA 26 - FUNÇÕES DE PERTINÊNCIA PARA O EXEMPLO DO CAMINHÃO.....	82
FIGURA 27 - ATIVAÇÃO DA REGRA $R^{5.1}$ E AS SUAS SAÍDAS	84
FIGURA 28 - ATIVAÇÃO DA REGRA $R^{5.2}$ E AS SUAS SAÍDAS	84
FIGURA 29 - ATIVAÇÃO DA REGRA $R^{6.2}$ E AS SUAS SAÍDAS	85
FIGURA 30 - CONJUNTOS NEBULOSOS UTILIZANDO T-CONORMAP/ INFERÊNCIAS:(A)MÍNIMO E (B)PRODUTO85	
FIGURA 31 - (A) I -CORTE E (B) a -CORTE EM CONJUNTOS NEBULOSOS.....	86
FIGURA 32 - EXEMPLO PARA O QUAL O DENEBUGADOR MEIO DO MÁXIMO NÃO FAZ SENTIDO...86	86
FIGURA 33 - PROCESSO INDUTIVO.....	90
FIGURA 34 - PROCESSO DE GENERALIZAÇÃO	91
FIGURA 35 - PROCESSO DE ESPECIALIZAÇÃO	92
FIGURA 36 - PROCESSO DE ANALOGIA.....	93
FIGURA 37 - VOLTA ATRÁS DIRECIONADA À INDEPENDÊNCIA	99
FIGURA 38 - ARQUITETURA SIMPLIFICADA DO EXPERT SINTA.....	104
FIGURA 39 - BARRA DE FERRAMENTAS	105
FIGURA 40 - JANELA “KNOWLEDGE-IN-A-BOX” (KIB).....	105
FIGURA 41 - BARRA DE FERRAMENTAS	106
FIGURA 42 - CRIANDO VARIÁVEIS	106
FIGURA 43 - DEFININDO VARIÁVEIS-OBJETIVO.....	107
FIGURA 44 - CAIXA DE DIÁLOGO “NOVA REGRA”	109
FIGURA 45 - CRIANDO REGRAS.....	109
FIGURA 46 - CRIANDO PREMISSA(S) DA REGRA.....	109
FIGURA 47 - CRIANDO CONCLUSÃO DA REGRA.....	110
FIGURA 48 - EDITANDO PREMISSA DA REGRA	110
FIGURA 49 - INTERFACE COM O USUÁRIO.....	111
FIGURA 50 - DEFININDO A INTERFACE COM O USUÁRIO	111
FIGURA 51 - CAIXA DE DIÁLOGO “INFORMAÇÕES SOBRE A BASE”	112
FIGURA 52 - DEFININDO TÓPICOS DE AJUDA PARA VALORES POSSÍVEIS DAS VARIÁVEIS-OBJETIVO	113

FIGURA 53 - DEFININDO O MODO DE AVALIAÇÃO DAS PREMISSAS PELA MÁQUINA DE INFERÊNCIA.....	114
FIGURA 54 - DEFININDO FATORES DE CONFIANÇA.....	114
FIGURA 55 - DEFININDO PERMISSÕES.....	115
FIGURA 56 - CRIANDO SENHA	116
FIGURA 57 - A BARRA DE FERRAMENTAS PARA EXECUÇÃO DE CONSULTAS.....	118
FIGURA 58 - OS ITENS DE MENU PARA A EXECUÇÃO DE CONSULTAS.....	118
FIGURA 59 - UMA ABERTURA TÍPICA DE UM SISTEMA ESPECIALISTA	119
FIGURA 60 - UMA PERGUNTA REALIZADA PELO SISTEMA A AO USUÁRIO.....	119
FIGURA 61 - O DEPURADOR.....	120
FIGURA 62 - CAIXA DE DIÁLOGO “ADICIONAR VARIÁVEIS”.....	121
FIGURA 63 - A LISTA DE VARIÁVEIS ACOMPANHADAS.....	121
FIGURA 64 - OS RESULTADOS ATINGIDOS.....	122
FIGURA 65 - A ÁRVORE DE PESQUISA	123
FIGURA 66 - TODOS OS VALORES ATINGIDOS PELO SISTEMA ESPECIALISTA	124
FIGURA 67 - A BASE DE CONHECIMENTO.....	124

LISTA DE TABELAS

TABELA 01 - COMPARAÇÃO ENTRE O CONHECIMENTO HUMANO E O CONHECIMENTO ARTIFICIAL	5
TABELA 02 - REGRAS PARA O EXEMPLO DO CAMINHÃO.....	83

LISTA DE ABREVIATURAS

BAM	Bidirectional Associative Memory
CAI	Inteligência Artificial Conexionista
CD	Dependência Conceitual
DOS	Disc Operating System
Fbf	Fórmulas bem formuladas
GPEB	Grupo de Pesquisas em Engenharia Biomédica
HES	Sistema Especialista Híbrido
HU	Hospital Universitário
IA	Inteligência Artificial
IAC	Inteligência Artificial Conexionista
IQ	Quociente de Inteligência
KRL	Linguagem de Programação de IA
LES	Lupus Eritematoso Sistêmico
LIFO	Last In First Out
LISP	Linguagem de Programação de IA
PLANNER	Linguagem de Programação
PPGCC	Programa de Pós-Graduação em Ciências da Computação
PROLOG	Linguagem de Programação de IA
PS	Pressão Sistólica
RNA	Rede Neural Artificial
SAI	Inteligência Artificial Simbólica
SCDM	Sistema de Controle de Pacientes Portadores de Diabete Mellitus
SE	Sistema Especialista
SEC	Sistema Especialista Conexionista
SEPRISCA	Sistema Especialista para Primeiros Socorros para Cães
SES	Sistemas Especialistas Simbólicos
UFSC	Universidade Federal de Santa Catarina

RESUMO

Neste trabalho foi desenvolvido um Sistema Especialista para Primeiros Socorros para Cães (SEPRISCA) que servirá não só como um “especialista virtual” aos proprietários de cães, mas também, como uma “memória adicional” aos médicos veterinários, interagindo com o usuário numa linguagem natural de perguntas e respostas, auxiliando-os nos primeiros socorros, de forma a amenizar ou resolver o problema.

Primeiramente serão abordados conceitos de Inteligência Artificial, incluindo Sistemas Especialistas (será conceituado de que se trata); Representação do Conhecimento, Mecanismos de Raciocínio.

Após, sobre Sistemas Especialistas; serão apresentadas: Ferramentas de Implementação e o sistema SEPRISCA que foi desenvolvido desde a criação de regras até a consulta.

Por fim, faz-se uma breve análise dos resultados obtidos como término desse trabalho, ressaltando os pontos mais significativos encontrados no decorrer de seu desenvolvimento.

O médico veterinário em seu trabalho utiliza uma série de instrumentos indo desde o termômetro e estetoscópio até novos aparelhos de ultrassom sem lesar a sensível audição dos animais. Pretende-se que o presente trabalho seja o protótipo de um sistema computadorizado que possa ser útil à decisão médica veterinária e forneça aos proprietários uma impressão diagnóstica que os permita esperar a chegada do veterinário, eventualmente ocupado em outro atendimento.

ABSTRACT

In this work a First Aid Specialist system for Dogs was developed (SEPRISCA) that will serve not only as a " virtual " specialist for the dogs' owners, but also, as an " additional " memory to the veterinary doctors, interacting with the user in a natural language of questions and answers, helping them in the First Aid, in a way to soften or to solve the problem.

Firstly, concepts about Artificial Intelligence will be approached, including Specialist Systems (it will be said what it means); the knowledge representation, the reasoning mechanisms.

After, about Specialist Systems; it will be presented implement tools and the SEPRISCA system that was developed from the creation of the rules to the consultation.

Finally, it is made a brief analysis of the obtained results as an end of this work, emphasizing the most significant points found in the elapse of its development.

The veterinary doctor in his work uses a series of instruments going from the thermometer and stethoscope to new ultrasound equipment without harming the sensitive animals 'audition. It is intended that the present work is the prototype of a computerized system that can be useful to the veterinary medical decision and can supply to the owners a diagnostic impression that allows them to wait for the vet's arrival, eventually busy in another attendance.

I - INTRODUÇÃO

1.1 - MOTIVAÇÃO

A medicina veterinária é uma profissão curiosa. Espera-se que o veterinário trate de praticamente toda emergência que surja, envolvendo qualquer espécie de animal. Essa é uma das satisfações e desafios da profissão.

Ao mesmo tempo, a medicina veterinária é uma ocupação altamente especializada; o aumento constante do volume de conhecimentos humanos induz a uma redução do campo de competência de cada indivíduo. Exige-se grande especialização para um indivíduo exercer uma atividade profissional de alto nível. Alguns veterinários concentram-se em determinadas espécies -gatos, cães, pássaros, cavalos, por exemplo. Outros especializam-se em campos específicos como ortopedia, oftalmologia ou dermatologia. A medicina emergencial e o atendimento de casos graves é um desses campos.

A evolução das ciências e das técnicas tendem a suprimir a noção do especialista interdisciplinar, mas se rende a freqüentes, e mais complexas, interações entre especialistas.

Muitas vezes os problemas reais são abordados por uma análise de grupo, onde a tomada de decisão, depende de um diagnóstico, da capacitação e treinamento de pessoal. Esta decisão depende de múltiplas avaliações que são funções da competência dos vários peritos em campos específicos. Os sistemas especialistas podem auxiliar na solução desses problemas.

Sistemas Especialistas procuram capturar e simular o conhecimento de especialistas humanos. Estes softwares são desenvolvidos para tratar de problemas complexos do mundo real que necessitem a interpretação e a análise de especialistas

humanos e que ao mesmo tempo cheguem, a conclusões e decisões que o especialista humano chegaria se estivesse se defrontando com os mesmos problemas.

Esses programas computacionais utilizam-se de regras de inferência sobre uma grande base de conhecimentos, sobre sintomas e tratamentos possíveis, para que possam identificar o problema e posteriormente oferecer um tratamento adequado, auxiliando à tomada de decisões.

Neste trabalho pretende-se desenvolver um Sistema Especialista para Primeiros Socorros para Cães, tendo como resultado; conhecimento, reconhecimento e retorno financeiro. A admiração por cães é um dos pontos fortes desta escolha.

O software terá sua aplicação em clínicas veterinárias, visando auxiliar o veterinário no diagnóstico e, na sua ausência ou ocupação, seus assistentes. Servirá como um “especialista virtual”, aos proprietários de cães, auxiliando-os nos primeiros socorros, de forma a amenizar ou resolver o problema, fornecendo-os uma impressão diagnóstica que os permita esperar a chegada do veterinário, eventualmente ocupado em outro atendimento.

1.2 - OBJETIVOS

1.2.1 - Objetivo Geral

Análise, desenvolvimento e implementação de um protótipo do sistema especialista “Sistema Especialista para Primeiros Socorros para Cães”.

1.2.2 - Objetivos Específicos

- Estudar a estrutura e funcionamento de sistemas especialistas;
- Estudar a linguagem de programação “Delphi”;
- Estudar características, comportamento específicos do animal, emergências (primeiros socorros), doenças;
- Conhecer as ferramentas “shells” existentes e escolher a adequada;
- Definição do projeto do sistema;

Desenvolvimento do sistema especialista.

II - SISTEMAS ESPECIALISTAS

2.1 - HISTÓRICO

Nos anos 50, os pesquisadores já haviam estabelecido as fundações da Inteligência Artificial (IA), incluindo Lógica Matemática e Teoria das Funções Recursivas, guiando a formulação de processamento de listas e da própria linguagem LISP que fornece um interpretador para desenvolver expressões simbólicas recursivas. Tais capacidades suportaram o surgimento de sistemas práticos de computação simbólica. Ao mesmo tempo, emergiram computadores interativos tornando possíveis ambientes computacionais para desenvolvimento e depuração de programas incrementais. Aproximadamente nesta mesma ocasião, psicólogos cognitivos - estudantes da forma de pensar humana - criaram caminhos padrão do processo de investigação do raciocínio, modelando o aparente processo de tomada de decisão em termos de regras de produção condicionais.

Nos anos 60, os pesquisadores de Inteligência Artificial tentaram simular o complexo processo do pensamento procurando métodos gerais para resolver uma ampla classe de problemas. Então durante a década de 70 concentraram esforços em técnicas como Representação - isto é, modo de formular o problema de maneira a tornar sua solução mais fácil e como controlá-la inteligentemente dentro da capacidade de memória do computador.

Somente no final da década fizeram a descoberta mais importante: o poder do programa em resolver problemas depende mais do conhecimento que possui do que do formalismo ou esquema de inferência empregado. Esta realização levou ao desenvolvimento de programas de computador de propósito particular, sistemas que são peritos em alguma área limitada. Estes programas são chamados "Sistemas Especialistas".

No campo de IA aplicada, após uma década de trabalho, emergiram três subcampos principais: Sistemas Especialistas, Linguagem Natural e Robótica - incluindo visão, fala e locomoção. Os sistemas especialistas começaram a surgir comercialmente entre 1980 e 1981. A primeira companhia formada exclusivamente para produzir sistemas especialistas foi a Intelli Genetics, no campo de engenharia genética e com técnicos oriundos do Projeto de Programação Heurística da Universidade de Stanford.

Assim os sistemas especialistas após terem se desenvolvido por 15 anos como mera curiosidade de IA aplicada em laboratórios de pesquisa, tornaram-se alvo de significativos esforços de desenvolvimento, tanto técnicos como comerciais. Recentemente, muitas organizações têm explorado esta tecnologia, ampliando suas pesquisas e começando a adaptar suas atividades para tanto.

Embora sistemas especialistas e peritos reais possam em alguns casos desempenhar tarefas idênticas, as características de ambos são criticamente diversas. Mesmo havendo algumas vantagens evidentes dos sistemas especialistas, eles não poderão substituir os peritos em todas as situações devido a algumas limitações inerentes. A tabela 01 resume esta comparação:

TABELA 01 - COMPARAÇÃO ENTRE O CONHECIMENTO HUMANO E O CONHECIMENTO ARTIFICIAL

CONHECIMENTO HUMANO	CONHECIMENTO ARTIFICIAL
Perecível	Permanente
Difícil de Transferir	Fácil de Transferir
Difícil de Documentar	Fácil de Documentar
Imprevisível	Consistente
Caro	Razoável
Discriminatório	Imparcial
Social	Individualizado
Criativo	Sem Inspiração
Adaptável	Inflexível
Enfoque Amplo	Enfoque Restrito
Baseado em Senso Comum	Técnico

2.2 - ALGUNS SE DESENVOLVIDOS NA UNIVERSIDADE FEDERAL DE SANTA CATARINA (UFSC)

Não se pretende fazer um histórico completo, apenas será mostrado alguns trabalhos desenvolvidos nessa Universidade, tomando por base os departamentos de

Ciências da Computação, Engenharia Elétrica e Engenharia de Produção e Sistemas, para não alongá-lo muito. Entretanto, sabe-se que outros departamentos vêm realizando trabalhos nessa área como, por exemplo, a Engenharia Civil.

Inicialmente, serão citados alguns trabalhos do Programa de Pós-Graduação em Ciência da Computação (PPGCC), ao qual este trabalho está vinculado.

Maria Aparecida Fernandes Almeida: defendeu sua dissertação de Mestrado em setembro de 1999, com o título de “Aprender Atividade Inteligente: E se esta Inteligência for Parcialmente Artificial?” [2].

Este trabalho analisa as perspectivas da IA no processo de Ensino-Aprendizagem com o computador. São observados alguns impactos da aplicação do computador no processo de ensino perante a sociedade contemporânea.

Carlos Efrain Stein: defendeu sua dissertação de Mestrado em agosto de 2000, com o título “Sistema Especialista Probabilístico: Base de Conhecimento Dinâmica” [70].

Rudimar Luís Scaranto Dazzi, defendeu sua dissertação de Mestrado em setembro de 1999, com o título “Sistemas Especialistas Conexionista Implementados por Redes Diretas e Bidirecionais” [19].

Este trabalho apresenta a implementação de Sistemas Especialistas Conexionistas (SEC), os quais utilizam a topologia de Redes Neurais Artificiais (RNA) Diretas. Após a apresentação dos conceitos básicos de Inteligência Artificial e suas abordagens simbólica e conexionista, apresentar-se-ão os fundamentos dos SEC.

A análise dos resultados do aprendizado, com a inserção de novos exemplos para o treinamento de redes diretas, será mostrado com o suporte de gráfico de acompanhamento, que apresenta a curva de desempenho da rede, representada pelos seguintes eixos: número de exemplos utilizados no treinamento pelo percentual de acertos obtidos nos testes.

Apresenta-se, também, os resultados obtidos com a tentativa de implementação da rede BAM que, em virtude de problemas de instabilidade encontrados e demonstrados no desenvolvimento desse, tornou-se impossível obter as conclusões, bem como os testes.

Por fim, faz-se uma breve análise dos resultados obtidos ao término desse trabalho, ressaltando os pontos mais significativos encontrados no decorrer de seu desenvolvimento.

Cabe citar aqui, os livros escritos por professores desse departamento, relacionados à área em questão:

Inteligência Artificial escrito por Renato A. Rabuske, publicado em 1995 [60].

Inteligência Artificial no Limiar do Século XXI, escrito por Jorge M. Barreto, em 1997 [6].

Programa de Pós-Graduação em Engenharia Elétrica no Grupo de Pesquisa em Engenharia Biomédica (GPEB). Aqui os estudos começaram, com o artigo escrito por Lima e Barreto, intitulado Inteligência Artificial em 1987 [46]. A partir desse artigo, vieram outros estudos e diversos trabalhos defendidos, pode-se citar os trabalhos de:

Marlise Vidal Montello: defendeu sua dissertação de Mestrado em fevereiro de 1999, com o título de “Sistema Especialista para Predições e Complicações Cardiovasculares Integrado a um Sistema de Controle de Pacientes Portadores de Diabetes Mellitus” [54].

Este trabalho tem por objetivo a implementação de um sistema de gerenciamento de dados, o SCDM (Sistema de Controle de Pacientes Portadores de Diabetes Mellitus), capaz de gerir todo o processo de atendimento a pacientes diabéticos realizado no Ambulatório de Endocrinologia do Hospital Universitário da Universidade Federal de Santa Catarina de forma a torná-lo mais ágil, eficiente e confiável. Desenvolveu-se um módulo especialista que, através de técnicas de Inteligência Artificial, é capaz de identificar se um determinado paciente apresenta ou não algum tipo de predisposição para desenvolver complicações cardiovasculares baseado na Análise dos Fatores de Risco.

Lourdes Mattos Brasil: defendeu sua tese de Doutorado em fevereiro de 1999, com o título de “Proposta de Arquitetura para Sistema Especialista Híbrido e a Correspondente Metodologia de Aquisição do Conhecimento” [11].

Este trabalho tem como meta propor uma metodologia para o desenvolvimento de um SE usando uma arquitetura híbrida. Este SE tem, como característica principal, a

capacidade de aprender a extrair conhecimento a partir de uma base de conhecimento inicial e de um conjunto de exemplos. Desta forma, espera-se contribuir para a solução de um dos problemas de IA, que consiste na extração de conhecimento do especialista de domínio e que, nos sistemas simbólicos, é conhecido como uma das etapas da Aquisição de Conhecimento.

Para mostrar a importância que os SE vem tendo dentro da UFSC, podemos apresentar também o trabalho desenvolvido no Programa de Pós-Graduação em Medicina Interna do Hospital Universitário (HU), onde o médico Li Shih Min defendeu sua tese, denominada “Sistema Baseado em Conhecimento para Detecção e Classificação de Crises Epilépticas” [52].

2.3 - ALGUMAS PUBLICAÇÕES DE SE DESENVOLVIDOS NA UNIVERSIDADE FEDERAL DE SANTA CATARINA (UFSC)

Rudimar Luís Scaranto Dazzi, Jorge Muniz Barreto, Silvia Modesto Nassar: “Sistemas Especialistas Conexionistas: Implementação por Redes Diretas e Bidirecionais”. Rev. Alcanse, ISSN 1413-2591, Ano VIII, v. 1, p. 107-115, 2001 [20].

Este trabalho apresenta a implementação de Sistemas Especialistas Conexionistas (SEC), os quais utilizam a topologia de Redes Neurais Artificiais (RNA) Diretas. Após a apresentação dos conceitos básicos de Inteligência Artificial e suas abordagens simbólica e conexionista, apresentar-se-ão os fundamentos dos SEC.

A análise dos resultados do aprendizado, com a inserção de novos exemplos para o treinamento de redes diretas, será mostrado com o suporte de gráfico de acompanhamento, que apresenta a curva de desempenho da rede, representada pelos seguintes eixos: número de exemplos utilizados no treinamento pelo percentual de acertos obtidos nos testes.

Apresenta-se, também, os resultados obtidos com a tentativa de implementação da rede BAM que, em virtude de problemas de instabilidade encontrados e demonstrados no desenvolvimento desse, tornou-se impossível obter as conclusões, bem como os testes.

Por fim, faz-se uma breve análise dos resultados obtidos ao término desse trabalho, ressaltando os pontos mais significativos encontrados no decorrer de seu desenvolvimento.

Lourdes Mattos Brasil, F. M. Azevedo, Jorge Muniz Barreto. Expert System used Hybridism among Symbolic and Conexionist Paradigms, Fuzzy Logic and Genetic Algorithms. Proceedings of the International Symposium on Medical Informatics and Fuzzy Technology (MIF'99), Hanoi, Vietnam, p. 347-354, 1999 [13] e,

Lourdes Mattos Brasil, F. M. Azevedo, Jorge Muniz Barreto. Hybrid Expert System for Decision Support in the Medical Area. In: Medical and Biological Engineering and Computing, Viena, Áustria, v. 37, n. 2, p. 738-739, 1999 [14].

O processo de aquisição de conhecimento consiste em extrair e representar conhecimento de domínio especialista.

Neste trabalho, um dos objetivos é minimizar as dificuldades intrínsecas do processo de aquisição de conhecimento. Outro objetivo, é proporcionar um Sistema Especialista Híbrido para minimizar os problemas de aquisição de conhecimento usando uma nova metodologia.

Construir esta arquitetura híbrida tem conduzido-nos a usar muitas ferramentas: Paradigma Simbólico, Paradigma Conexionista, Lógica Fuzzy e, Algoritmos Genéticos. Outro objetivo é apresentar dois novos algoritmos, o primeiro é um algoritmo de aprendizagem para ser aplicado para redes neurais feed-forward fuzzy, bem como otimizar a complexidade de problemas da topologia de rede. O segundo é extrair regras fuzzy de uma rede neural fuzzy treinada. O algoritmo de aprendizagem foi inspirado no clássico algoritmo Back-Propagation. Possui algumas variações que se deve ao tipo de rede usada. O algoritmo de extração de regras fuzzy também possui algumas particularidades. Assim a metodologia desenvolvida para sistemas especialistas híbridos como ambos algoritmos foram testados através de problemas reais e fictícios.

Jorge Muniz Barreto. Inteligência Artificial e Engenharia Biomédica: Casamento Perfeito ou Amantes Eternos?. XVII Congresso Brasileiro de Engenharia Biomédica(CBEB2000), Florianópolis, setembro 2000 [5].

Este artigo descreve um caso de simbiose entre Engenharia Biomédica e a Inteligência Artificial (IA). Inicia-se com um curto histórico da IA, ressaltando sua evolução no Grupo de Pesquisas em Engenharia Biomédica (GPEB) da UFSC. Mostra como a Inteligência Artificial foi progressivamente entrando na biomédica, inicialmente com os sistemas especialistas, para depois utilizar redes neurais e a matemática nebulosa. Ao longo da apresentação, algumas vezes informal, apresenta dificuldades e sucessos que ajudam a responder a pergunta “Casamento Perfeito ou Amantes Eternos?”. O trabalho conclui com conjecturas sobre futuras aplicações, tais como bases de dados inteligentes.

Jovelino Falqueto, Walter C. Lima, Paulo S. S. Borges, Jorge Muniz Barreto. The Measurement of Artificial Intelligence: An IQ for Machines?. Proceedings of the IASTED International Conference on Modelling, Simulation and Control. Innsbruck, Áustria, v. 1, p. 409-413, february 19-22, 2001. Acta Press, Anaheim, Calgary, Zurich, ISBN 0-88986-316-4, ISSN 1025-8973 [25].

A Medida de Inteligência Artificial: Um QI para Máquinas?.

Falando uma vez sobre as possibilidades de pensar criar " máquinas ", Turing disse corretamente que esta tarefa deveria começar definindo do que é compreendido como " máquinas " e do que é " pensa ". Relativo à perseguição de modelar inteligência, duas avenidas grandes foram abertas por investigadoras em quase a mesma época: Inteligência Artificial Simbólica - SAI - e Inteligência Artificial Conexionista - CAI - baseadas respectivamente em símbolos e regras e em neurônios artificiais. Parece que o tempo é vir começar pensando (e agindo) para estabelecer um padrão de comparação que poderia objetivamente contar como distante nós fomos ao longo da estrada para construir sempre melhores sistemas de IA. Inventando um Quociente de Inteligência IQ - para máquinas ou qualquer sistema inteligente seria, talvez, um avanço, mas infelizmente, a história do desenvolvimento de técnicas para medir IQ humano, a primeira fonte conferiu achar aplicações para AI, pontos para uma zona muito difusa.

Admitindo aquela possibilidade, nós apresentamos algumas conjeturas. Por exemplo, introduzindo algum métrico para avaliar a redundância das regras de um sistema especialista, ou a eficiência de uma determinada topologia numa rede neural poderia trazer novas perspicácias em ordenar paradigmas de AI e indica que são o mais promissor.

Rudimar Luís Scaranto Dazzi, Jorge Muniz Barreto. *Sistemas Especialistas Conexionistas em Reumatologia*. UFSC, Florianópolis [18].

Lógicas paraconsistentes e/ou para completas preservando todas as características da lógica clássica que não violem as qualidades deviantes da negação pretendidas por cada uma delas.

Lourdes Mattos Brasil, F. M. Azevedo, Jorge Muniz Barreto. *A Hybrid Expert System for the Diagnosis of Epileptic Crisis*. *Artificial Intelligence in Medicine*. Elsevier, n. 585, p. 1-7, 2000 [12].

Um Sistema Especialista Híbrido para a Diagnose de Crises Epiléptica.

Este presente trabalho, um Sistema Especialista Híbrido (HES), pretendeu minimizar alguns problemas complexos penetrantes para conhecimento da engenharia tais como: o processo de elicitación do conhecimento, conhecido como o gargalo de garrafa de sistemas especialistas; a escolha de um modelo para representação de conhecimento para codificar o raciocínio humano: o número de neurônios na camada escondida e a topologia usada na aproximação conexionista; a dificuldade para extrair uma explicação da rede. Também são sugeridos dois algoritmos aplicados para desenvolvimento de HES. Um deles é usado para treinar a rede neural fuzzy e o outro obter explicações de como a rede neural fuzzy atingiu uma conclusão. Um estudo de caso está presente (por exemplo crise epiléptica) com a inclusão de definição de problema e simulações. Os resultados também são discutidos.

2.4 - CONCEITOS

Sistema: conjunto de elementos, materiais ou ideais, entre os quais se possa encontrar ou definir alguma relação.

Especialista: pessoa que se consagra com particular interesse e cuidado a certo estudo. Conhecedor, perito.

Sistemas especialistas são sistemas que solucionam problemas que são resolvíveis apenas por pessoas especialistas (que acumularam conhecimento exigido) na resolução destes problemas. São sistemas computacionais criados para resolver problemas e que devem apresentar um comportamento semelhante a um especialista em um determinado domínio, cujo conhecimento utilizado é fornecido por pessoas que são especialistas naquele domínio.

Um sistema especialista é um programa inteligente de computador que se utiliza de métodos inferenciais para a resolução de problemas técnicos e altamente especializados. Por utilizar-se da Inteligência Artificial, um ramo da computação que estuda a capacidade de uma máquina raciocinar e aprender como um ser humano, os sistemas especialistas interagem com seu usuário numa linguagem natural de perguntas e respostas, sugerindo e auxiliando na solução de problemas complexos.

Programas de computador que tentam resolver problemas que os seres humanos resolveriam emulando o raciocínio de um especialista, aplicando conhecimentos específicos e inferências são ditos “Sistemas Especialistas” [43].

O sistema convencional é baseado em um algoritmo, emite um resultado final correto e processa um volume de dados de maneira repetitiva enquanto que um sistema especialista é baseado em uma busca heurística e trabalha com problemas para os quais não existe uma solução convencional organizada de forma algorítmica disponível ou é muito demorada [43].

Um sistema especialista é aquele que é projetado e desenvolvido para atender a uma aplicação determinada e limitada do conhecimento humano. É capaz de emitir uma decisão, apoiado em conhecimento justificado, a partir de uma base de informações, tal qual um especialista de determinada área do conhecimento humano [43].

Para tomar uma decisão sobre um determinado assunto, um especialista o faz a partir de fatos que encontra e de hipóteses que formula, buscando em sua memória um conhecimento prévio armazenado durante anos, no período de sua formação e no decorrer de sua vida profissional, sobre esses fatos e hipóteses. E o faz de acordo com a

sua experiência, isto é, com o seu conhecimento acumulado sobre o assunto e, com esses fatos e hipóteses, emite a decisão.

Durante o processo de raciocínio, vai verificando qual a importância dos fatos que encontra comparando-os com as informações já contidas no seu conhecimento acumulado sobre esses fatos e hipóteses. Neste processo, vai formulando novas hipóteses e verificando novos fatos, que influenciaram no processo de raciocínio. Este raciocínio é sempre baseado no conhecimento prévio acumulado. Um especialista com esse processo de raciocínio pode não chegar a uma decisão se os fatos de que dispõe para aplicar o seu conhecimento prévio não forem suficientes. Pode, por este motivo, inclusive chegar a uma conclusão errada, mas este erro é justificado em função dos fatos que encontrou e do seu conhecimento acumulado previamente [43].

Segundo Feigenbaum (apud Harmon [35]) um sistema especialista (SE) "é um programa inteligente de computador que usa conhecimentos e procedimentos inferenciais, para resolver problemas que são bastante difíceis, de forma a requererem para sua solução, muita perícia humana. O conhecimento necessário para atuar a esse nível, mais os procedimentos inferenciais empregados, pode considerar-se um modelo da perícia aos melhores profissionais do ramo. O conhecimento de um sistema especialista consiste em fatos e heurísticas. Os fatos constituem um corpo de informação que é largamente compartilhado, publicamente disponível e geralmente aceito pelos especialistas em um campo. As heurísticas são, em sua maioria privadas, regras poucos discutidas, de bom discernimento (regras de raciocínio plausível, regras de boa conjectura), que caracterizam a tomada de decisão a nível de especialista na área. O nível de desempenho de um sistema especialista é função principalmente do tamanho e da qualidade do banco de conhecimento que possui."

Um sistema especialista deve, além de inferir conclusões, ter capacidade de aprender novos conhecimentos e, desse modo, melhorar o seu desempenho de raciocínio, e a qualidade de suas decisões [43].

Para Henri Farreny [26] um sistema só é considerado especialista se ele possui as seguintes características:

Uma linguagem de expressão dos conhecimentos fornecidos pelos especialistas;

Uma base de conhecimentos, para armazenar o conhecimento específico de determinada aplicação, que pode ser diretamente fornecido por um especialista, ou acumulado pelo sistema ao fim dos experimentos;

Um motor de inferência, programa relativamente geral que explora o conhecimento da base precedente, considerando-a como fonte de informações.

Os sistemas especialistas são construções de software que os peritos em campos específicos enriquecem com seu conhecimento, capacitando um computador a auxiliá-lo num processo de tomada de decisão. A intenção em usar conhecimento simbólico para simular o comportamento dos especialistas humanos é justamente desenvolver programas que possibilitem a utilização dos conhecimentos dos especialistas através de uma máquina que permita o armazenamento e o seqüenciamento de informações e a auto-aprendizagem.

Assim Dimitri N. Chorafas [16] coloca que: "o computador interage com as perguntas do usuário e chega a uma conclusão baseado nas respostas. O usuário pode perguntar ao computador por que ele quer certa informação e o SE explicará sua necessidade dos dados e como eles serão utilizados. O mais importante é que o SE dirá como chegou a suas conclusões; não dará somente conselhos, mas também justificará a opinião que oferece."

Jacques Quibel [59] em seus estudos diz que: "desenvolver um SE é aplicar uma "engenharia" do conhecimento: a construção de uma informática lógica, refere-se à fatos e idéias e a capacidade de tratá-los, os quais receberá do exterior, para deduzir um diagnóstico, uma precognização, ou, de fato, uma decisão."

2.5 - QUANDO OS SISTEMAS ESPECIALISTAS SÃO UTILIZADOS

De um modo geral, sempre que um problema não pode ser algoritmizado, ou sua solução conduza a um processamento muito demorado, os sistemas especialistas podem ser uma saída, pois possuem o seu mecanismo apoiado em processos heurísticos.

Preservar e transmitir o conhecimento de um especialista humano em uma determinada área.

Um sistema especialista não é influenciado por elementos externos a ele, como ocorre com o especialista humano, para as mesmas condições deverá fornecer sempre o mesmo conjunto de decisões.

2.6 - A EFICÁCIA DOS SISTEMAS ESPECIALISTAS

Para que um sistema especialista seja eficaz, as pessoas têm de ser capazes de interagir com ele facilmente. Para facilitar esta interação os sistemas devem ser capazes de:

Explicar seu raciocínio: conseqüentemente o processo de raciocínio deve proceder em etapas compreensíveis em que o metaconhecimento suficiente (conhecimento sobre o processo de raciocínio) esteja disponível para que as explicações dessas etapas possam ser geradas [42];

Adquirir conhecimento novo e modificar o conhecimento antigo: como o conhecimento pode ser aumentado e/ou alterado, torna-se importante então separar a base de conhecimento do conjunto de operadores do sistema [42].

2.7 - PRINCIPAIS BENEFÍCIOS DA UTILIZAÇÃO DOS SISTEMAS ESPECIALISTAS

Velocidade na determinação dos problemas;

A decisão está fundamentada em uma base de conhecimento;

Segurança. A confiabilidade do software é um importante fator. A possibilidade de exame da montagem da regras geradoras de uma conclusão, permite encontrar as inferências indutivas e suas justificações. Quando uma pessoa faz o mesmo trabalho intelectual, mesmo que tenha uma perícia e um certo grau de inteligência, chega-se que ela esquece ou distorce a realidade, cometendo assim um erro. Se o software comete uma falha, esta pode ser consertada, pois foi introduzida durante a concepção e assim a validação não é feita corretamente e o resultado não é apresentado. Ele é geralmente mais

fidedigno que o homem. Ele não sofre, como a mente humana, a influência da psique, das condições externas, da fadiga...;

Exige pequeno número de pessoas para interagir com o sistema;

Estabilidade;

Dependência decrescente de pessoal específico. Graças ao sistema especialista, põe-se o problema para o computador em lugar do especialista, às vezes indisponível no momento requerido. Sua experiência fica assim confortavelmente explorável, para a disposição de um número maior de pessoas, isso sem se cansar.

Flexibilidade;

Integração de ferramentas;

Evita interpretação humana de regras operacionais. Softwares construído com linguagens clássicas do processo de dados, requerem um conhecimento rígido, encaixados uns aos outros, em uma determinada ordem, enquanto as regras de uso de um sistema especialista são flexíveis, graças ao seu motor de inferência;

A separação da informação e o seu tratamento, facilita a restrição à regras inadequadas ou obsoletas, e a sua modificação inclui a evolução e a introdução de novas regras. O conhecimento considerado no fato geralmente pode ser atualizado;

Um sistema especialista oferece atrativos como: calma, infatigabilidade, tempo de vida quase eterno se atualizado, ao contrário do indivíduo: rebelde, frágil, fatigável (então menos eficiente às vezes) e mortal;

Captura conhecimentos que serão utilizados no futuro. A experiência dos especialistas são perdidas com o tempo, no momento em que eles se especializam em outros problemas ou trocam de trabalho. Um sistema especialista nunca esquece um procedimento, nem mesmo em seus mínimos detalhes [41];

Reduz os custos com funcionários especializados. Com sistemas especialistas uma tarefa de grande complexidade pode ser realizada por funcionários que

não sejam *experts*, mas funcionários especializados apenas. Com isso, os especialistas podem ficar dedicados a tarefas que realmente sejam de altíssima complexidade;

Sistemas especialistas podem freqüentemente fazer melhor que um especialista. sistemas especialistas podem combinar a capacidade de um especialista com a velocidade e precisão de uma máquina. Eles podem fazer uma análise detalhada e completa de uma situação que dificilmente um humano seria capaz de fazer. Segundo, se um sistema foi bem estruturado, ele não cometerá erros que eventualmente um humano pode cometer. Finalmente, um sistema especialista seria capaz de fazer tarefas detalhadas e repetitivas - que um especialista talvez as desprezasse, em uma situação em que as chances de resolver o problema são muito pequenas [41];

Podem ser a maior ajuda no treinamento de pessoal. Os sistemas especialistas podem fornecer informações detalhadas - de diversos níveis, sobre como foi procedido para solucionar um determinado problema. Esta pode ser a maior ajuda no treinamento de pessoas inexperientes. Em adição, existem sistemas especialistas construídos especialmente para prover treinamento. Neste caso, até didática eles possuem [41].

2.8 - PROBLEMAS ENFRENTADOS PELOS SISTEMAS ESPECIALISTAS ATUAIS

Fragilidade. Como os sistemas especialistas somente têm acesso a conhecimento altamente específicos do seu domínio, não possuem conhecimentos mais genéricos quando a necessidade surge [42];

Falta de metachecimento. Geralmente não possuem conhecimentos sofisticados sobre sua própria operação, portanto não conseguem raciocinar sobre seu próprio escopo e restrições. A aquisição do conhecimento continua sendo um dos maiores obstáculos à aplicação de tecnologia dos sistemas especialistas a novos domínios [42];

Validação. A medição do desempenho de sistemas especialistas é muito difícil porque não sabemos quantificar o uso de conhecimento [42];

Os sistemas especialistas também podem gerar problemas. Com a introdução da automação, os trabalhadores podem se sentir inseguros -eles podem achar que irão perder o emprego (e isto realmente acontece). Esta situação pode ser evitada com uma migração lenta e responsável do sistema humano para o sistema máquina [41];

Um outro problema pode surgir quanto a política. Sistemas especialistas não possuem senso de política, podendo produzir resultados embaraçosos a alguém. Portanto, nessas situações, é necessário subordinar o sistema a um analista humano, que interage com ele e desvia a situação se necessário [41];

2.9 - CLASSIFICAÇÃO DE SISTEMAS ESPECIALISTAS

Podemos classificar os sistemas especialistas quanto às características do seu funcionamento. De um modo geral, tais categorias são:

Interpretação: são sistemas que inferem descrições de situações à partir da observação de fatos fazendo uma análise de dados e procurando determinar as relações e seus significados [40]. Ex: Fotos de satélite (para sensoriamento remoto).

Diagnóstico: são sistemas que detectam falhas oriundas da interpretação de dados. A análise dessas falhas pode conduzir à uma conclusão diferente da simples interpretação de dados. Detectam os problemas mascarados por falhas dos equipamentos e falhas do próprio diagnóstico, que este não detectou por ter falhado. Estes sistemas já têm embutidos o sistema de interpretação de dados. Ex: Diagnóstico médico.

Monitoração: interpreta as observações de sinais sobre o comportamento monitorado. Tem de verificar continuamente um determinado comportamento em limites preestabelecidos, sinalizando quando forem requeridas intervenções para o sucesso da execução. Um sinal poderá ser interpretado de maneiras diferentes, de acordo com a situação global percebida naquele momento, e a interpretação varia de acordo com os fatos que o sistema percebe a cada momento. Ex: Supervisão de processos industriais.

Predição: a partir de uma modelagem de dados do passado e do presente, este sistema permite uma determinada previsão do futuro. Como ele baseia sua solução na análise do comportamento dos dados recebidos no passado, de ter mecanismos para verificar os vários futuros possíveis, a partir da análise do comportamento desses dados, fazendo uso de raciocínios hipotéticos e verificando a tendência de acordo com a variação dos dados de entrada.

Planejamento: o sistema prepara um programa de iniciativas a serem tomadas para se atingir um determinado objetivo. São estabelecidas etapas e subetapas e, em caso de etapas conflitantes, são definidas as prioridades. O princípio de funcionamento, em alguns casos, é por tentativas de soluções, cabendo a análise mais profunda ao especialista que trabalha com esse sistema. Ex: Planejamento da trajetória de robôs.

Projeto: é um sistema capaz de justificar a alternativa tomada para o projeto final, e de fazer uso dessa justificativa para alternativas futuras [40].

Depuração: trata-se de sistemas que possuem mecanismos para fornecerem soluções para o mau funcionamento provocado por distorções de dados. Provê, de maneira automática, verificações nas diversas partes, incluindo mecanismos para ir validando cada etapa necessária em um processo qualquer.

Reparo: este sistema desenvolve e executa planos para administrar os reparos verificados na etapa de diagnóstico. Um sistema especialista para reparos segue um plano para administrar alguma solução encontrada em uma etapa do diagnóstico. Ex: manutenção de aeronaves.

Instrução: o sistema de instrução tem um mecanismo para verificar e corrigir o comportamento do aprendizado dos estudantes. Seu funcionamento consiste em ir interagindo com o treinando, em alguns casos apresentando uma pequena explicação e, a partir daí, ir sugerindo situações para serem analisadas pelo treinando. Dependendo do comportamento deste, se vai aumentando a complexidade das situações e encaminhando o assunto, de maneira didática, até o nível intelectual do treinamento. Ex: treinamento de operadores de processos.

Controle: é um sistema que governa o comportamento geral de outros sistemas (não apenas de computação). É o mais completo, de um modo geral, pois deve

interpretar os fatos de uma situação atual, verificando os dados passados e fazendo uma predição do futuro. Apresenta os diagnósticos de possíveis problemas, formulando um plano ótimo para sua correção. Este plano de correção é executado e monitorado para que o objetivo seja alcançado [40]. Ex: Controle de processos industriais.

2.10 - FERRAMENTAS PARA IMPLEMENTAÇÃO DE SE

2.10.1 - SHELL de Sistemas Especialistas

Inicialmente, os sistemas especialistas eram criados do nada. Mas devido ao fato de os sistemas serem construídos como um conjunto de representações declarativas (em sua maioria, regras) combinadas com um interpretador dessas representações, era possível separar o interpretador do conhecimento específico do domínio da aplicação e assim criar um sistema que podia ser usado para elaborar novos sistemas especialistas através da adição de novos conhecimentos, correspondentes ao novo domínio do problema. Estes interpretadores resultantes são chamados de “SHELLs”, que servem de base para muitos dos sistemas especialistas que estão sendo desenvolvidos. O nome shells vem do fato de tornar transparente as dificuldades inerentes à implementação de uma aplicação em Inteligência Artificial: escolha da representação de conhecimento, do método de busca, ferramentas para encontrar erros, etc.

A construção de sistemas especialistas envolvem investimentos em profissionais de diversas áreas, implicando em riscos maiores para a produção. Uma solução para este problema é o uso de ferramentas capazes de prototipar, avaliar e implementar o projeto de um sistema, os chamados “SHELLs”. Procura-se, assim, diminuir a necessidade de recursos necessários, entre eles o tempo envolvido e tenta-se conciliar a alta produtividade e a versatilidade de ferramentas qualificadas.

Um shell é uma ferramenta genérica para implementação de bases de conhecimento. Que de tal modo, o desenvolvedor não se preocupa com a máquina de inferência, apenas com o conjunto de regras que modelam o conhecimento desejado.

Normalmente um SHELL inclui:

Ferramentas para introduzir conhecimento na base de conhecimento;

Um ou vários motores de Inferência;

Mecanismos para tratar imprecisão;

Mecanismos para tratar incertezas;

Mecanismos de seguir passo a passo o funcionamento do programa de modo a encontrar possíveis erros;

Uma comunicação homem-máquina.

Exemplos de SHELLs para a construção de sistemas especialistas:

Expert Sinta: sua finalidade é embutir dentro de seus próprios ambientes, meios para construção de interfaces gráficas com o usuário final, além de mecanismos de troca de dados com outras aplicações.

KappaPC: permite escrever aplicações em um ambiente gráfico e gerar códigos padrão ANSI C. Ele é uma ferramenta de auxílio na construção de sistemas baseados em conhecimento, usando frames, regras de produção, e programação orientada a objetos.

VP-Expert: sistema baseado em regras. Usa o raciocínio para frente e possui mecanismos embutidos para o tratamento de imprecisão com o paradigma dos fatores de confiança.

Advisor: ambiente de desenvolvimento baseado na tecnologia de regras de negócio, construído 100% Java.

Expert: conhecido também como Nexpert, é uma ferramenta de desenvolvimento para ambientes Windows e Unix, que fornece a gerentes e desenvolvedores a capacidade de integrar a experiência do negócio.

Clips: é a abreviação de 'C Language Integrated Production System', foi projetado pela NASA/Johnson Space Center.

FuzzyClips: utiliza-se da teoria e dos cálculos da Lógica Fuzzy, para representar o conhecimento.

2.11 - CICLO DE VIDA DE SISTEMAS ESPECIALISTAS

2.11.1 - Fases do Desenvolvimento de um Sistema Especialista

Nos sistemas especialistas, o desenvolvimento, minimiza bastante a separação em etapas, as quais formalmente continuam a ser consideradas como: Identificação, Conceituação, Formalização, Implementação, Teste e Avaliação, e Revisão.

2.11.1.1 - Fase de Identificação

Nesta fase devem ser identificados os participantes do projeto, os recursos envolvidos, as características do problema e os objetivos a atingir.

Na identificação dos participantes deve ser determinado em primeiro lugar “o dono” do sistema, uma pessoa que pertença à administração da empresa, tendo a última palavra que concerne a aspectos ligados ao sistema. Depois identifica-se o engenheiro de conhecimento, o qual terá auxiliares, se o sistema for muito grande.

O tempo e a dedicação da equipe que o desenvolverá são características tocantes ao esforço de desenvolvimento.

2.11.1.1.1 - Identificação dos Recursos

Compreende em primeiro lugar, a identificação das fontes de conhecimento, como o especialista, os livros, as revistas, os fichários, etc. A entrevista com um especialista no domínio da aplicação é necessária para elucidar o conhecimento especialista, que é então traduzido em regras. Depois que o sistema inicial estiver pronto, ele precisa ser interativamente refinado até aproximar-se do nível de desempenho de um especialista.

Compreende em segundo lugar, a delimitação do tempo, estabelecendo um cronograma básico.

Compreende, em terceiro lugar, a identificação dos recursos “computacionais”, especificando as máquinas a serem usadas, máquinas a adquirir, “softwares”, etc.

Por último, não menos importante, compreende, ainda, a identificação dos recursos financeiros envolvidos.

2.11.1.1.2 - Identificação das Características do Problema

Deverão ser considerados tópicos como os seguintes, apresentados em Hayes-Roth [37].

Que classes de problemas o sistema deverá resolver?

Como podem ser caracterizados ou definidos estes problemas?

Quais os principais subproblemas?

Quais são os dados?

Quais os termos importantes e suas interrelações?

Que é uma solução?

Que aspectos o perito humano acha importante na solução?

Que situações costumam impedir a solução?

Como estas soluções afetarão um sistema especialista?

Respondidas estas perguntas, serão descritos juntamente com o especialista os principais elementos do problema, em outras palavras, formalizar o problema e identificar o objetivo. A identificação do objetivo é necessária, pois pode impor restrições adicionais ao problema.

2.11.1.2 - Fase da Conceituação

Consiste em definir a base conceitual do sistema especialista. O engenheiro de conhecimento e o especialista decidirão quais os recursos básicos necessários para desenvolver o problema (conceitos, relações, mecanismos de controle) e estabelecerão também o grau de refinamento que será usado na representação do conhecimento.

A fase da conceituação não dispensa uma olhada para frente, verificando quais as possíveis representações e ferramentas que poderão servir de base para implementação. Isto, porém, deverá ser feito com cuidado para evitar que os passos subseqüentes sejam viciados, impondo-se restrições em função da ferramenta que possivelmente será adotada.

2.11.1.3 - Fase da Formalização

Envolve a expressão de conceitos e de relações-chaves, de uma maneira formal, identificando estruturas de suporte para sua representação e armazenamento. Se estas estruturas forem parte integrante de alguma ferramenta existente, poder-se-á utilizá-la para construção do sistema.

O engenheiro de conhecimento, que agora predomina as ações, nesta fase deve prender a atenção em três aspectos, a saber: o espaço de hipóteses, o modelo subjacente e as características dos dados.

O primeiro destes aspectos implica em estabelecer o quanto serão refinados os conceitos e como eles se interligam. Deve-se, também, estabelecer as características que acompanharão estes conceitos.

O modelo subjacente determina a forma como as soluções serão geradas.

As características dos dados envolvem a definição de aspectos como: tipos, precisão, consistência, volume e formas de aquisição.

2.11.1.4 - Fase da Implementação

Esta fase se consuma com a edição do conhecimento e a feitura dos programas que o processam, quando não for feita opção por alguma ferramenta já existente. Os programas deverão respeitar conteúdo das estruturas, forma de raciocínio e integração do todo (através da estratégia de controle). Isto, em geral, não é fácil de ser conseguido, razão pela qual os programas são revistos muitas e muitas vezes.

O objetivo central desta etapa deve ser o teste da adequação da forma de representação escolhida e das estruturas de suporte adotadas.

2.11.1.5 - Fase do Teste e Avaliação

O sistema especialista deverá ser testado e avaliado freqüentemente, desde a implementação inicial do sistema. Deverá ser levado em consideração o desempenho e a utilidade. O teste poderá por a descoberto falhas de representação, exigindo, então revisão.

O teste é sempre um momento muito importante porque ele abrange não só o sistema em si, como um corpo que está sendo trabalhado, mas também leva em consideração a razão de ser deste sistema.

A avaliação é um passo que, com frequência, recebe pouca atenção, porque não é uma tarefa excitante e ainda há pouco esforço de pesquisa nesta área. Mas é a única forma de salvaguardar o sucesso de um empreendimento. A avaliação pode ser balizada por diversas questões, a seguir consideradas, que facilitarão a acentuação do que é importante.

Que avaliar: é preciso avaliar a correção das respostas, a explicação, a interação homem-máquina, a aceitação do sistema especialista, o cronograma, a qualidade da interação com o usuário, a eficiência, a precisão e a credibilidade do programa.

Quando avaliar: deve-se avaliar em todas as fases e estágios do desenvolvimento do sistema especialista, desde o projeto de alto nível até o “marketing”.

Como avaliar: comparando o conhecimento do sistema com o do especialista, o que, em geral, não proporciona resultados confiáveis, ou comparando a forma de o sistema especialista resolver o problema, confrontando-o com o humano.

Uma forma de avaliação a ser considerada é a participação de pessoas que não ajudaram no desenvolvimento do sistema, não conhecendo-o, portanto, em pormenores. Este tipo de avaliação poderá retratar a sensação dos futuros usuários do sistema.

Outros aspectos a considerar: é importante escolher padrões de avaliação realistas, por exemplo, levando em consideração o estado da arte da área. Com poucos experimentos também não se deve tirar conclusões gerais. Uma bateria de testes, pré-estabelecida, também pode ser tendenciosa, uma vez que, possivelmente, ela poderá ter sido seguida como paradigma de desenvolvimento.

Não se deve esquecer de avaliar a elegância do sistema especialista. Respostas “secas”, embora corretas, podem entusiasmar bem menos que respostas bem

apresentadas. O uso diário da computação ensina que a elegância nunca deve ser subestimada.

Enfim, não se deve esquecer que os sistemas especialistas constituem uma área nova, e, portanto, que há muitas ferramentas ainda não consolidadas. As próprias linguagens de programação, com frequência, apresentam diferenças em resultados. Os esquemas de inferência, embutidos em algumas ferramentas, também não são fáceis de serem entendidos plenamente, o que poderá, então ser uma fonte adicional de erros.

2.11.1.6 - Fase da Revisão

É um processo continuado, acompanhando o desenvolvimento do sistema. Consiste em revisar o sistema, especialmente para alterar e melhorar aspectos observados na fase de avaliação. A revisão deve ser feita, tendo em mente o escopo definido para o sistema. Se nesta fase houve mudanças significativas, possivelmente haverá necessidade de uma reavaliação geral do sistema.

III - ARQUITETURA E SISTEMAS ESPECIALISTAS SIMBÓLICOS

3.1 - SISTEMAS ESPECIALISTAS SIMBÓLICOS (SES)

São os SE baseados no simbolismo, ou seja, base de conhecimento, motor de inferência, regras, etc.

Dois pontos importantes a serem considerados são:

Os fatos, dados, sintomas ou características do domínio do problema (as informações relevantes para que se possa resolver o problema). Ex: Quando seu carro não funciona corretamente, você descreve para o mecânico (especialista), o que está ocorrendo (Motor está falhando, temperatura alta, etc). E as regras que o especialista usa para resolver o problema. As analogias que ele faz para chegar a uma possível resposta para o problema em questão.

Isso é cuidadosamente analisado pelo projetista do sistema (*engenheiro de conhecimento*), o qual irá inserir na base de dados do sistema todo esse conhecimento, toda essa experiência retirada do especialista. O trabalho do *engenheiro de conhecimento* é extremamente importante, uma vez que o especialista têm a capacidade de resolver problemas difíceis, explicar os resultados obtidos, aprender, reestruturar o conhecimento e determinar as suas características relevantes. Porém, geralmente o especialista não consegue explicar o seu modo de raciocínio de uma maneira analítica [17]. A maioria dos SES que nasceram nos anos 60 eram organizados ao redor dessa estrutura.

3.2 - SISTEMAS ESPECIALISTAS CONEXIONISTAS (SEC)

O *conexionismo* é uma das duas grandes linhas de pesquisa da IA e tem por objetivo investigar a possibilidade de simulação de comportamentos inteligentes através de modelos baseados na estrutura e funcionamento do cérebro humano [10].

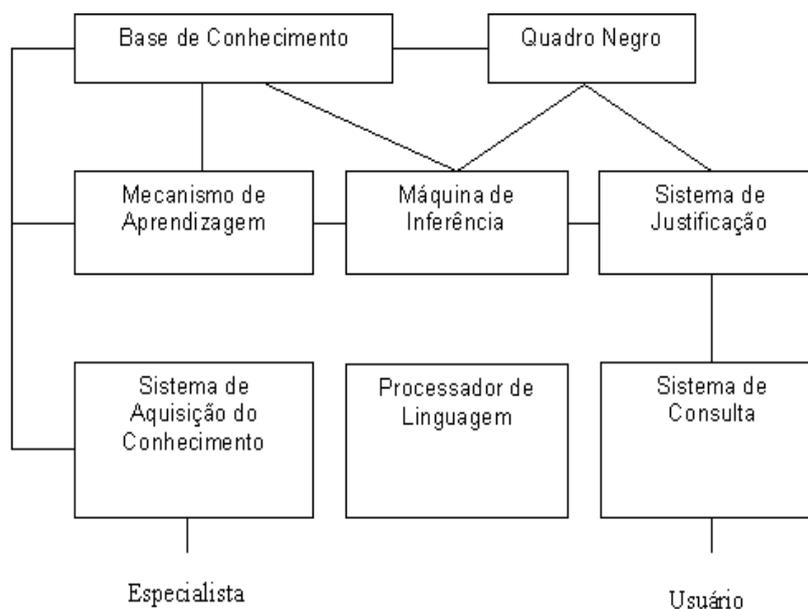
Aqui o foco de interesse são as RNA, com suas mais diversas topologias.

Os SEC são especialmente indicados, quando o domínio do problema em questão não é bem definido, quando existem muitas exceções as regras, de forma que não se consiga definir com precisão as regras para a extração do conhecimento. Como os SEC, são baseados em exemplos e não em regras, pode-se resolver problemas sem o conhecimento de como tomar as decisões, bastando para isso ter exemplos de casos reais. Barreto [6] divide o domínio de atuação dos SE em dois tipos: o das coisas criadas pelo homem e as criadas pela natureza. Quando se tem os feitos pelo homem, se conhece bem o domínio e é fácil conseguir todos os detalhes do assunto. Mas nos naturais o sistema é desconhecido ou pouco conhecido e seu funcionamento pode ser conhecido através do uso de modelos e a estrutura do sistema conhecida através de exemplos. Neste segundo caso, fica claro a indicação do uso de SEC para resolver o problema.

3.3 - UMA COMPARAÇÃO SIMPLIFICADA ENTRE A ESTRUTURA DE UM SES E DE UM SEC

Um sistema especialista simbólico possui os componentes que estão apresentados na figura 01.

FIGURA 01 - ARQUITETURA DE UM SISTEMA ESPECIALISTA SIMBÓLICO (SES)



Base de Conhecimento

Segundo Feigenbaum [27], “A potência de um sistema especialista deriva do conhecimento que ele possui e não dos formalismos e esquemas específicos que ele emprega”.

A base de conhecimento, é um elemento permanente, mas específico de um sistema especialista. Contém conhecimento, sob a forma de regras de produção, quadros, redes semânticas, ou outra forma qualquer. Poder-se-ia dizer que a base de conhecimentos contém um somatório de fatos, de heurísticas e de crenças. Ela é criativa, capaz de certos tipos de controle sobre si mesma, podendo até suprir algumas informações ausentes. É especialmente esta última característica que a distingue das tradicionais bases de dados.

Máquina de Inferência

Máquina de inferência ou motor de inferência é um elemento permanente, que pode inclusive ser reutilizado por vários sistemas especialistas. É a parte responsável pela busca das regras da base de conhecimento para serem avaliadas, direcionando o processo de inferência. O conhecimento deve estar preparado para uma boa interpretação e os objetos devem estar em uma determinada ordem representados por uma árvore de contexto. Basicamente a máquina de inferência é dividido em tarefas que são: selecionar, buscar, avaliar e procurar. Resumindo as tarefas acima, podemos dizer que as regras necessárias para se chegar a uma meta devem ser buscadas na base de conhecimento. Essas regras serão colocadas no quadro negro, sendo que as regras já existentes só serão avaliadas depois das mais recentes. A ordem de avaliação no quadro negro obedece a uma estrutura do tipo pilha com o objetivo de atingir a meta mais recente. A regra continuará sendo avaliada enquanto as condições da premissa forem verdadeiras, caso contrário a regra será eliminada, a meta estabelecida desempilhada e uma nova regra será carregada.

Quando um valor de um parâmetro em um determinado contexto não é conhecido e não se encontra nas estruturas de pilha, deve-se então procurar novas informações na base de conhecimento, buscando novas regras ou perguntando diretamente ao usuário.

Quadro Negro

Também conhecido como rascunho ou memória de trabalho tem sua vida útil durante o curso de uma consulta e está vinculada a uma consulta concreta. É uma área de memória usada para fazer avaliações das regras que são recuperadas da base de conhecimento para se chegar a uma solução. As informações são gravadas e apagadas em um processo de inferência até se chegar a solução desejada.

Sistema de Justificação

Chamada em Forsyth [31] de “a janela humana”, é uma capacidade de questionamento, fornecida ao usuário, seja para repetir uma dedução efetuada, seja para responder a outras questões que o sistema especificamente permita. A justificação é um requisito obrigatório nos sistemas especialistas, tendo, geralmente, capacidade de responder às seguintes perguntas:

Como chegou a esta conclusão?

Porque chegou a esta conclusão?

Por que não chegou a tal outra conclusão?

Mecanismo de Aprendizagem

A aprendizagem comum se dá de diversas formas:

Análise estatística de dados (heurística);

Tentativa e erro (experiências);

Leituras, palestras, etc;

Troca de experiências com outras pessoas.

Fundamentalmente, verifica-se que o aprendizado vem do processo de experiência, e de seus resultados experimentais.

A capacidade de aprender, no ser humano, é o resultado de um conjunto de habilidades: capacidade de generalizar, de induzir, de fazer analogias e de receber instrução.

Os sistemas especialistas devem ser capazes de aprender e fazer crescer o seu conhecimento básico sobre o assunto. Esta capacidade de aprender recebe o nome técnico de *protopeiria*.

O usual é existir um engenheiro de conhecimento que prepara o conhecimento para ser armazenado em uma forma apropriada, fornecendo as explicações necessárias, dos conceitos utilizados. O ideal é que o conhecimento possa ser adquirido diretamente pelo sistema especialista.

Uma das formas de aprendizagem dos sistemas especialistas é através de textos. Um programa captura palavras chaves em um parágrafo do texto, podendo formatá-lo para um formato especial de armazenamento, para representação desse conhecimento (implementação mais eficiente e confiável com processador de linguagem natural).

O aprendizado também pode ser feito a partir de conclusões sobre a massa de informações mantidas pelo sistema especialista. Ele mantém um banco de casos resolvidos, isto é, a cada conclusão guarda os fatos que pesaram sobre a decisão, após estar criticado por um especialista da área. O aprendizado é feito por comparação de dados por um módulo do sistema especialista que coloca a nova regra na base de conhecimento, à medida que a massa de dados cresce, obedecendo o formato adequado.

Outra forma de aprendizado se dá pela interação direta com o especialista. Como em uma relação professor-aluno, o computador absorve o conhecimento através de uma interface adequada (editor inteligente).

Fundamentalmente, verifica-se que o aprendizado vem do processo de experiência, e de seus resultados experimentais.

O Sistema de Aquisição de Conhecimento

É um módulo que permite ao sistema ampliar e alterar seu conhecimento. Aparece, geralmente, munido de recursos para trabalhar o conhecimento (editores, ordenadores, classificadores, etc) que auxiliam na difícil tarefa de extraí-lo e aproveitá-lo adequadamente. Em muitos sistemas é a única forma de aprendizado.

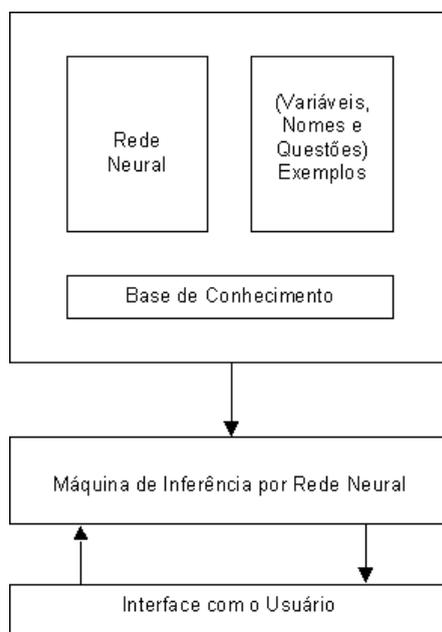
Sistema de Consulta

O usuário é, em geral, alguém que não participou da elaboração do sistema especialista, sendo, portanto, natural que não conheça as estruturas sustentadoras do sistema e, que, provavelmente, não esteja familiarizado com as formas de representação do conhecimento adotadas.

Para que os potenciais usuários possam acessar com proveito e sem maiores dificuldades o sistema especialista, é preciso, então, muni-lo de recursos para consulta, que são módulos explícitos ou implícitos ao sistema. Para contornar a linguagem técnica da tecnologia “computacional” e da engenharia do conhecimento, estes módulos estabelecem uma linguagem orientada para o problema, podendo ser um subconjunto da língua nativa. Na figura 01, o módulo “processador de linguagem” reúne estas características. Ele não foi ligado a qualquer módulo, sendo, porém, requerido todas as vezes que o usuário estiver ativo.

Um sistema especialista conexionista possui os componentes apresentados na figura 02.

FIGURA 02 - ARQUITETURA DE UM SISTEMA ESPECIALISTA CONEXIONISTA (SEC)



Base de Conhecimento

É composta dos exemplos usados pela rede neural para o seu treinamento (onde a rede adquire conhecimento). Aqui entra a fase de treinamento (redes diretas) ou a fase de inserção dos pesos na matriz de conexões (memórias associativas).

Máquina de Inferência por Rede Neural

É o próprio mecanismo da rede neural, o qual recebe as entradas vindas da interface com o usuário, interage e retorna a resposta (os neurônios ativados da saída), para a mesma.

Interface com o Usuário

É uma parte do sistema utilizada para estabelecer um meio de comunicação entre o SE e o usuário. Pode-se considerar a mesma explicação dos SES.

Outra diferença a se considerar, é o fato das BAM poderem utilizar dois tipos de conhecimento. O conhecimento do especialista e o conhecimento de casos (exemplos) simultaneamente. As redes diretas não utilizam o conhecimento causal do especialista diretamente. Para o treinamento das redes diretas utiliza-se apenas os exemplos. Acredita-se com isso, que os SE implementados por BAM tenham um desempenho superior, conforme resultados apresentados por Azevedo [21].

3.4 - CONHECIMENTO

As pessoas, desde o nascimento, ou mesmo antes, vão acumulando conhecimentos que lhes permitirão agir de modo a mostrar que são seres inteligentes. Isto comumente é denominado de aprendizado.

Geralmente pensa-se o aprendizado de forma muito simplificada, como a acumulação de conhecimento. Porém, envolve, de fato, um ciclo completo de processamento da informação, que vai desde a coleta do conhecimento pelos sentidos, até seu armazenamento definitivo no cérebro. Este processo pode ser descrito como compreendendo os seguintes passos:

- 1) Ocorre o estímulo sensorial;
- 2) Ocorre a retenção do estímulo por pouco tempo;
- 3) A informação é trabalhada pela memória de trabalho;
- 4) Ocorre a resposta;
- 5) A informação é registrada, ou não, na memória de longa duração.

Apesar disto, a pessoa não capta todos o conhecimento que a realidade do mundo lhe proporciona, nem acumula definitivamente toda a informação percebida e a possível de ser captada. Também, pessoas diferentes, perante a mesma realidade, não captam e nem entendem a mesma coisa. Pessoas com mais conhecimentos costumam captar melhor as coisas que são inerentes a sua área de conhecimento que as com menos conhecimento acumulado.

Então depreende-se que trabalhar com conhecimentos não é tarefa fácil, especialmente quando tentamos usar mecanismos artificiais, como os computadores; em que a manipulação do conhecimento exige, antes, formas de representação.

Não se conhecendo exatamente como o cérebro humano trabalha, e também, dispondo-se somente de máquinas que essencialmente foram projetadas para lidar com números, é necessário, antes de tudo, pensar como o conhecimento é estruturado e como podemos guardá-lo e manipulá-lo no computador.

3.4.1 - Comparação entre Conhecimento Especializado Humano e Artificial

Um sistema especialista pode ser comparado com um programa gravado em videocassete enquanto que neste contexto um especialista se compararia com um programa ao vivo. Isto é, a utilização e reprodução, que compensa seu alto custo de implantação, pode levar vários anos. Os especialistas humanos, especialmente os melhores, tem altos salários, enquanto que sistemas especialistas tem o custo do software e de manutenções.

Por outro lado, para demonstrar que os especialistas não podem ser simplesmente substituídos, o conhecimento artificial tem atualmente algumas limitações. Uma delas é falta de criatividade. Um perito pode reorganizar informações e usá-las para sintetizar novos conhecimentos, podendo manusear eventos inesperados usando imaginação ou

novas abordagens, inclusive raciocínio por analogia de um outro domínio completamente diferente. Sistemas especialistas trabalham de maneira sem inspiração, rotineiramente.

Finalmente, os seres humanos - peritos ou não - possuem o conhecimento advindo do senso comum, que se constitui num largo espectro de conhecimento sobre o mundo, acumulado durante toda a sua vida e que permeia todas as suas decisões. Devido a enorme quantidade de conhecimento de senso comum, fatos de domínio público que todos sabem, torna-se difícil construir um programa inteligente, particularmente um sistema especialista.

Como um exemplo deste tipo de conhecimento, suponha que você examina um registro médico de um paciente que pese 7 quilos e tenha idade de 25 anos. Imediatamente você suspeitará de erro nos dados, não porque isoladamente sejam impossíveis, mas sua combinação virtualmente o será. Desconfiará que os dados foram acidentalmente invertidos. Um sistema especialista não detectaria este tipo de erro, a menos que disponha de uma tabela idade/peso (e seus limites de tolerância) para conferi-lo.

Senso comum inclui conhecimento sobre o que não se sabe assim como o que se conhece. Por exemplo, se você for perguntado como a televisão noticiou um fato ocorrido no ano passado, puxará pela memória para recuperar a informação. Se a pergunta fosse "como a TV soviética noticiou outro fato", imediatamente responderá que não sabe. Da mesma forma, se lhe perguntarem "como a TV brasileira informou a Proclamação da República", você dirá que tal não ocorreu pois não havia televisão naquela época. Quando um sistema especialista é questionado sobre informações que não disponha ou não existam, ele não detecta esta situação por não possuir senso comum. Então, iniciará exaustiva pesquisa nos seus fatos e regras para encontrar a solução. Daí, quando esta não for encontrada, pode julgar que é porque seu conhecimento está incompleto e solicita informação adicional para completar sua base de conhecimento. Por estas razões, sistemas especialistas são frequentemente utilizados no aconselhamento, como consultor ou ajuda para outro perito ou iniciante usar em algum problema.

Em contraste com sistemas avançados de processamento de dados, que automatizam algoritmicamente extensos volumes de dados, os sistemas especialistas ordinariamente trabalham com pequenas tarefas tipicamente desempenhadas por profissionais: interpretando, diagnosticando, planejando, escalonando e assim por diante. Para acompanhamento destas tarefas, os sistemas especialistas utilizam judiciosamente os dados e raciocínios com eles. Ao contrário do método usual em processamento de dados, os sistemas especialistas geralmente examinam uma extensa gama de possibilidades ou constroem soluções dinamicamente.

3.4.2 - Conhecimento e sua Representação

Segundo Fischler e Firschein [29], “conhecimento pode ser definido como a informação armazenada, ou métodos usados pela pessoa ou máquina para interpretar, prever e responder apropriadamente ao mundo exterior”.

É importante distinguir entre forma e conteúdo do conhecimento. Um texto que usa a linguagem natural, como recurso de representação, pode ter, também, seu conteúdo sintetizado através de outros recursos, como, por exemplo, a lógica dos predicados. O uso de uma ou outra forma de representação traz consigo resultados diferentes. O uso da linguagem natural pode, por exemplo, facilitar ao usuário um entendimento maior do conteúdo, enquanto a lógica dos predicados pode facilitar a manipulação “computacional” desta informação. Outro ponto de vista que pode auxiliar, ainda, a ressaltar a importância para distinguir entre forma e conteúdo no armazenamento do conhecimento é a necessidade do usuário. Por exemplo, por mais que um literato se esmere na descrição de uma coluna jônica, não conseguirá o mesmo efeito que uma simples fotografia, produzida por um amador qualquer.

Uma representação de um objeto, por sua vez, é a tradução deste objeto para um sistema, gerando uma imagem que pode ser trabalhada, relacionada com outras imagens, identificada e entendida.

Para as representações apresentarem algum interesse, são, também, necessárias funções capazes de efetuar o mapeamento entre a representação e o mundo real e vice-versa.

As funções necessárias para efetuar o mapeamento também não são tão simples de serem elaboradas. Há representações que admitem diversas interpretações, seja considerando a semântica envolvida, seja levando em conta o ponto de vista do usuário. Então, querendo-se dar um conhecimento genuíno à máquina, deve-se pensar em formas de representação que independam das intenções e propósitos do usuário. Só desta forma ter-se-ão elementos que a máquina possa considerar relevantes para “ela mesma”. O significado intrínseco torna-se aqui relevante.

3.4.2.1 - Porque Representar o Conhecimento?

Representa-se o conhecimento para posteriormente recuperá-lo, para raciocinar com ele e, para adquirir mais conhecimento. Isto permite caracterizar algumas funções mais comumente desempenhadas pela representação do conhecimento, quais sejam:

Função Dedutiva: a obtenção de novos resultados, a partir de dados previamente armazenados. O campo mais propício para este tipo de experimento é a matemática, especialmente o campo da prova de teoremas.

Função Consulta: grandes quantidade de informações podem gerar dificuldades no cruzamento delas. Responder questões, fazendo buscar na base de conhecimentos, é uma aplicação interessante.

Função Organização: sistemas “computacionais” não costumam funcionar adequadamente sobre informação não devidamente organizada. Organizar a informação ajuda a entendê-la, facilitando a intercomunicação entre formas diferentes de representação.

Função Interpretação: permite, através de comparações com modelos pré-estabelecidos e armazenados previamente, efetuar interpretações de uma possível realidade, dificilmente verificável de outra forma.

3.4.2.2 - Características de Boas Representações de Conhecimento

A coleta da informação que constitui o conhecimento deve ser feita em função dos objetivos a atingir. Nas representações não deve haver informação implícita, salvo a

inerente às estruturas de informação, que lhes servem de suporte. Estas estruturas podem ajudar a atribuir valores que são passados por herança.

Abaixo relacionadas um conjunto de qualidades, apresentadas em Winston [74]; que caracterizam as representações de conhecimento.

Boas representações explicitam as coisas importantes;

Revelam restrições naturais, facilitando algumas classes de computações;

São completas, podendo dizer tudo o que devem ser capazes de dizer;

São concisas, necessitando apenas de recursos mínimos e sendo ao mesmo tempo ainda eficientes quando efetuam inferências;

São transparentes, não usando criptografia, podendo-se entender o que querem dizer;

Facilitam a computação no sentido de que a informação pode ser armazenada e recuperada rapidamente;

Suprimem pormenores e informações raramente usadas, as quais são abordadas e recuperadas apenas quando necessárias;

São computáveis por um procedimento existente não exigindo desenvolvimento de programação exclusiva;

Permitem uma aquisição fácil do conhecimento e são legíveis pelo especialista, quando for o caso;

Permitem a aplicação dos mecanismos de inferência necessários;

Apresentam um acesso rápido e fácil ao conhecimento;

Permitem manter todo o conhecimento coerente, utilizando, se necessário, recursos para proteger o acesso.

3.4.3 - Principais Formas de Representação do Conhecimento

3.4.3.1 - Representação do Conhecimento em IAC

Como já foi dito, as informações se armazenam com intensidade de conexões sinápticas. Mas aqui cabe uma reflexão: existirá alguma razão para que a uma sinapse corresponda uma informação? Isto seria bem cômodo, ao menos para nossa compreensão, pois corresponderia ao que ocorre nos computadores com que temos tido mais contacto, quer utilizem DOS, WINDOWS, OS/2, LINUX ou MAC/OS. Nestes computadores têm-se uma memória dividida em endereços e cada informação é armazenada em uma posição de memória bem definida. Assim, para um computador com uma memória de disco rígido de 1,2G podemos por no máximo 1,2G informações distintas (ou menos se uma informação ocupa mais de um endereço).

E nas RNAs? Ora, as RNAs se inspiram na realidade biológica. Assim, apesar de uma representação localizada ser possível, ela é terrivelmente antieconômica e certamente a natureza escolheu outra solução. Supondo uma representação localizada, temos cerca de 10^{12} neurônios com uma média de 10^4 conexões entre neurônios ou seja cerca de 10^{16} sinapses (um erro por fator de 100 é razoável nesta afirmação devido às dificuldades de medida e à variabilidade biológica). Suponha-se ainda que a unidade de intensidade da conexão possa ser medida pelo número de moléculas neurotransmissoras liberadas para uma atividade determinada do neurônio excitador (isto é uma hipótese plausível, não uma realidade comprovada) e que possa variar em uma gama de 0 a 10000), dando ao total 10^{20} informações diferentes que podem ser armazenadas. Ou seja o fantástico número de 10 seguido de 20 zeros! Número enorme, mas muito pequeno se comparado com o número de coisas que um cérebro humano pode memorizar!

Mas, se a cada informação correspondesse uma configuração particular?. Neste caso seria como se o número de sinapses fosse o tamanho de uma palavra de computador, computador este trabalhando na base 10000 e não na base 2. Logo o número de coisas distintas que poderiam ser armazenadas será $10^4 \cdot 10^4 \cdot 10^4 \dots 10^4$ e isto repetido 10^{16} vezes! o que ultrapassa de muito qualquer coisa que se conheça na Natureza!.

3.4.3.2 - Representação do Conhecimento em IAS

A lógica pode ser usada para representar conhecimento. Seja, por exemplo, uma fbf do cálculo das proposições tal como:

$cor(gato, preto)$

Em Lógica esta fbf pode ter valor verdade ou falso, dependendo se o gato a que se refere a fbf é ou não preto.

De forma análoga, seja uma fbf do cálculo dos predicados, isto é envolvendo variáveis. Ela também pode ser usada para representar conhecimento. Seja por exemplo:

$\forall(x, y, z)(filho(x, y) \wedge filho(y, z) \Rightarrow neto(x, z))$

A representação do conhecimento usando lógica usa fbf da Lógica de Primeira Ordem e a todas elas é dado o valor de verdade “Verdade” formando uma base de fatos e de regras e constituindo a *base de conhecimentos*. Um mecanismo externo a esta base de conhecimentos irá manipulá-la, com regras de inferência (ex: modus ponens), para resolver o problema desejado.

3.4.3.2.1 - Sistemas de Produção

O termo “sistema de produção” é atualmente usado para descrever uma família de sistemas, que têm em comum o fato de serem constituídos de um conjunto de regras, que reúnem condições e ações. A condição (lado esquerdo, ou antecedente, ou premissa) é constituída por um padrão que determina a aplicabilidade de regra, enquanto a ação (lado direito, ou conseqüente) indica o que será realizado quando a regra for aplicada.

Um sistema de produção poderá ser formado por uma ou mais bases de regras, separadas segundo as conveniências de processamento. Complementa, ainda, o sistema de produção, uma estratégia de controle, e estabelecendo as prioridades em que as regras serão aplicadas, bem como critérios de desempate quando houver mais regras candidatas à aplicação a um só tempo. Este último aspecto denomina-se resolução de conflitos.

3.4.3.2.1.1 - Vantagens e Desvantagens das Regras de Produção

Os sistemas de produção são a forma de representação de conhecimento mais usada em IA. A causa disto reside no fato de ser natural ao humano usar o par “condição-ação” para raciocinar e decidir.

As vantagens dos sistemas de produção são:

Modularidade: as regras dos sistemas de produção podem ser consideradas, para efeito de manipulação, como peças independentes. Como os programas de IA quase sempre estão incompletos, novas regras podem ser acrescentadas ao conjunto já existente sem maiores preocupações. Esta é uma característica extremamente importante, pois o volume do conhecimento em geral é grande e a verificação de sua consistência, difícil. Para sistemas muito grandes a modularidade é um elemento que aumenta indiretamente a complexidade.

Naturalidade: considera-se a regra de produção uma forma natural de pensar a solução de problemas, por isso, o que deve ser feito em determinada circunstância é diretamente traduzido em regras de produção.

Uniformidade: num sistema de produção, todas as regras estão escritas seguindo o mesmo padrão. Esta padronização, por vezes, pode gerar alguns inconvenientes. Contudo, esta forma rígida de representação permite a que pessoas não familiarizadas com o sistema possam também analisar o seu conhecimento. Hoje, a grande maioria dos “softwares” que processam sistemas de produção, permitem sair destes ambientes para processar outras formas de apresentação do conhecimento que o usuário adotar, eliminando, em parte, as restrições que a uniformidade impõe.

3.4.3.2.1.2 - Desvantagens dos Sistemas de Produção

Opacidade: característica dos sistemas de produção resultante da modularidade e da uniformidade, fazendo com que seja difícil verificar a completeza destes sistemas, bem como verificar os possíveis fluxos de processamento. Consegue-se contornar um pouco este problema, se as regras forem separadas em subconjuntos, com o objetivo de clarear seu entendimento.

Ineficiência: é outra desvantagem resultante da modularidade e da uniformidade. A ineficiência dos sistemas de produção resulta, particularmente, do número de regras a combinar, e também do esforço de “matching” necessário ao suporte de execução das regras. “Matching” entende-se aqui como a verificação das regras que se aplicam ao estado do problema, bem como, a verificação de quais regras antecedem ou sucedem outra regra. Uma forma de diminuir a ineficiência é investir na ordenação apropriada das regras e formas de selecioná-las. Esta solução, no entanto, é “ad hoc”, variando bastante de sistema para sistema.

3.4.3.2.2 - Redes Semânticas

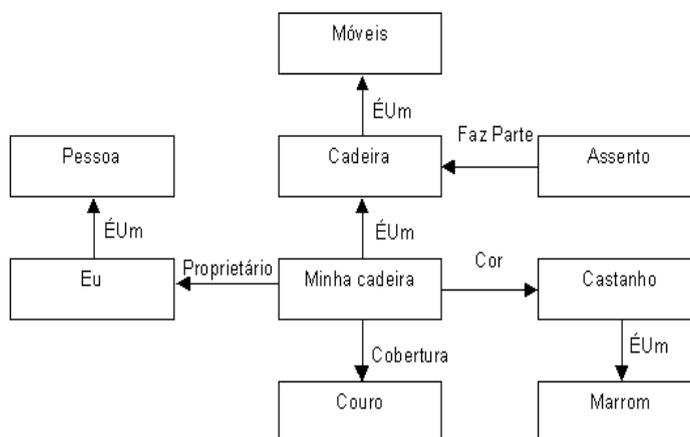
Basicamente rede semântica é representada por um grafo (dígrafo), com arestas e nós rotulados. A rotulação deve ser mnemônica, de sorte a ajudar a lembrar aquilo que a rede representa. Se a rotulação não conseguir traduzir a semântica, então não se está perante uma rede semântica.

Na rede semântica os nós (vértices) representam objetos, situações ou conceitos, sendo os elementos pertencentes à rede, enquanto que os arcos (arestas) exprimem as relações entre estes elementos. Esta descrição faz com que ela tenha algo em comum com a maioria das representações do conhecimento, as quais, em última análise, devem descrever objetos e suas inter-relações.

A semântica mais comum é a semântica descritiva, que procura mostrar através de linguagem natural o quanto a descrição se aproxima da realidade representada. Há, no entanto, outros tipos de semânticas com certo espectro de uso. Destacam-se a semântica procedural, que recorre a programas para exprimir seu significado, e a semântica de equivalência, que procura de alguma forma estabelecer equivalências entre a representação efetuada e uma outra forma de representar conhecimento. A semântica descritiva é mais atraente do que as outras, especialmente porque os seus recursos de representação são mais naturais. A semântica “procedural” exige, por exemplo, o entendimento de uma linguagem de programação.

Um fragmento de uma rede semântica típica é apresentado na figura 03.

FIGURA 03 - UMA REDE SEMÂNTICA



Normalmente são representadas utilizando algum tipo de estrutura de memória de valor de atributo. Assim, por exemplo, no LISP, cada nó seria um átomo, os elos seriam propriedades e os nós, nas outras extremidades dos elos, seriam os valores. A rede apresentada na figura 03 seria representada no LISP como é visto na figura 04.

FIGURA 04 - REPRESENTAÇÃO LISP DE UMA REDE SEMÂNTICA

LISTA DE PROPRIEDADES

ÁTOMO	((ÉUM MÓVEIS))
CADEIRA	((ÉUM CADEIRA))
	((COR CASTANHO))
	((COBERTURA COURO))
	((PROPRIETÁRIO EU))
EU	((ÉUM PESSOA))
CASTANHO	((ÉUM MARROM))
ASSENTO	((FAZPARTE CADEIRA))

Observe que cada propriedade armazena um elo de mão única, como o arco de MINHA-CADEIRA até EU. Para armazenar elos bidirecionais é necessário armazenar cada metade separadamente. Assim, se quiséssemos nos capacitar a responder à pergunta “O que possuo?” sem buscar a rede inteira, precisaríamos de arcos de EU para todos os nós que se ligam a EU através dos arcos POSSUIDOR. Como ir de MINHA-CADEIRA para EU e ir de EU para MINHA-CADEIRA têm significados diferentes, precisamos de outro tipo de arco, que podemos denominar POSSUIDO. É claro que é mais eficiente armazenar apenas arcos de mão única. Mas fica difícil formar inferências que vão na direção oposta. É preciso decidir que tipo de inferências será necessário para cada sistema, e depois projetar os elos adequadamente: este é o mesmo problema que os projetistas de qualquer base de dados deverão resolver quando tiverem de decidir quais campos indexar.

Até agora está claro que as redes semânticas podem ser utilizadas para representar relações que apareceriam como predicados de dois argumentos na lógica de predicados. Por exemplo, alguns dos arcos da figura 03 poderiam ser representados na lógica como:

ÉUM(cadeira,móvel)
 ÉUM(eu, pessoa)
 COBERTURA(minha-cadeira,couro)
 COR(minha-cadeira,castanho)

Mas o conhecimento expresso por outros predicados também pode ser expresso nas redes semânticas. Já vimos que muitos predicados de um argumento, na lógica, podem ser considerados como predicados de dois argumentos, usando alguns predicados de finalidade muito geral como ÉUM. Assim, por exemplo:

HOMEM(Marco)
 Poderia ser reescrito como:

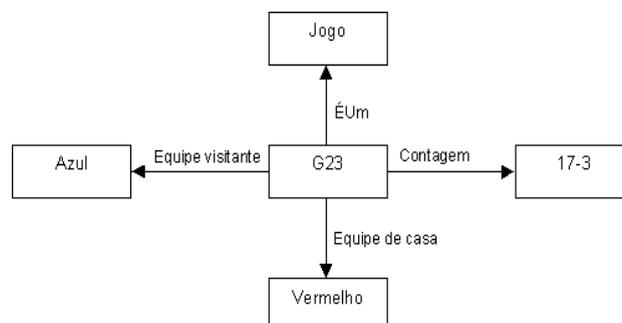
ÉUM(Marco,homem)
 tornando fácil assim representar uma rede semântica.

Predicados de três ou mais argumentos também podem ser convertidos para a forma binária, criando um novo objeto que represente a declaração inteira do predicado e depois introduzindo predicados binários para descrever a relação deste novo objeto com cada um dos argumentos originais. Por exemplo, suponhamos que sabemos que

CONTAGEM (vermelho azul(17 3))

Isto pode ser representado na rede semântica criando-se um nó que represente o jogo específico e depois relacionando cada uma das três peças de informação a ele. Fazer isto produz a rede apresentada na figura 05.

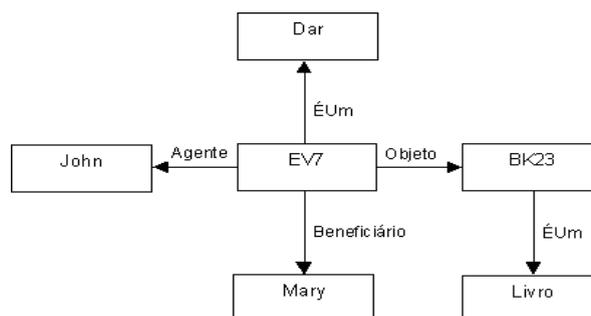
FIGURA 05 - UMA REDE SEMÂNTICA PARA UM PREDICADO DE N- ARGUMENTOS



Esta técnica é particularmente útil para representar o conteúdo de uma frase declarativa típica, que descreve diversos aspectos de um evento em particular. A sentença:

John deu o livro para Mary
poderia ser representada pela rede apresentada na figura 06.

FIGURA 06 - UMA REDE SEMÂNTICA REPRESENTANDO UMA FRASE



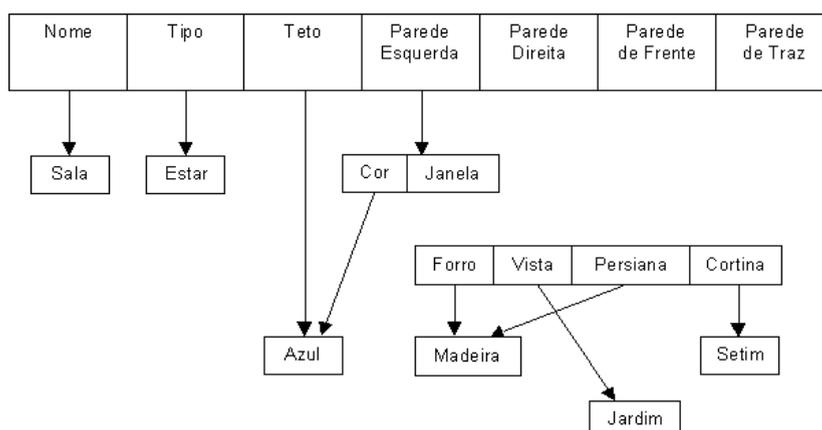
3.4.3.2.3 - Molduras (“Frames”)

Existe muita evidência de que as pessoas não analisam situações novas a partir do zero e depois constróem novas estruturas de conhecimento para descrever essas situações. Em vez disso, têm disponível na memória uma grande coleção de estruturas representando sua experiência anterior com objetos, localizações, situações e pessoas. Para analisar uma nova experiência, elas evocam estruturas apropriadas armazenadas e depois as preenchem com os detalhes do evento atual. Um mecanismo geral projetado para a representação computacional de tal conhecimento comum é a moldura. A palavra moldura foi aplicada a uma série de estruturas de representação de preenchimento. Em certo sentido, essas estruturas podem ser vistas como redes semânticas complexas; contudo, terão muita estrutura interna tipicamente projetada para torná-las úteis em tipos específicos de tarefas de solução de problemas.

Marvin Minsky, em 1975 [53] fez a conjectura que seria útil para armazenar conhecimentos, quebrá-las em pequenos conhecimentos quase elementares aos quais chamou “frames”, em português molduras. Molduras se tornaram populares principalmente quando se faz mister, armazenar conhecimento estereotipado em um determinado domínio.

Uma moldura é uma estrutura de dados que pode ser usada para representar qualquer entidade estereotípica real ou imaginária, uma situação ou um evento. As molduras podem ser agrupadas em seqüência para representar uma situação mutante ou ligadas hierarquicamente ou ainda em forma de uma rede.

FIGURA 07 - MOLDURA PARCIAL DE UMA SALA DE ESTAR



Uma moldura contém um objeto constituído por várias *campos* (“slots”) onde são colocadas as informações ligadas ao objeto. O conteúdo de tais campos são tipicamente os atributos do objeto particular. Entretanto, em adição aos valores armazenados para cada atributo, campos podem conter valores default, ponteiros para outras molduras e conjuntos de regras de procedimentos que podem ser implementadas.

Junto a cada moldura existem vários tipos de informações. Algumas destas informações são sobre como usar a moldura, outras sobre o que se pode esperar que aconteça a seguir e ainda sobre o que fazer se estas expectativas não forem confirmadas. O mecanismo representativo que possibilita o tipo de raciocínio baseado em expectativas é o campo, o local onde o conhecimento ajusta-se ao contexto criado pelo frame.

Uma característica deste tipo de representação de conhecimento é que cada campo pode ter qualquer número de procedimentos associados a ele. Além disso, os frames são úteis porque facilitam a inferência de fatos, ainda não observados, a respeito de situações novas.

Na figura 07 está representado parte do que seria a moldura correspondente a uma sala de estar. Note-se que alguns dos campos apontam para objetos, ou seja, têm valor fixado. Outros apontam para novas molduras, como no caso de parede.

As molduras são úteis para domínio de problema onde a forma e o conteúdo do dado desempenham um papel importante na solução do problema.

3.4.3.2.4 - Roteiros

O roteiro é uma estrutura que descreve uma seqüência estereotipada de eventos em um contexto em particular. O roteiro consiste em um conjunto de escaninhos. Associado ao escaninho poderá haver alguma informação a respeito do tipo de valores que ele poderá conter, bem como um valor de omissão a ser utilizado, se nenhuma outra informação estiver disponível.

Os eventos descritos num roteiro formam uma cadeia causal. O início da cadeia é o conjunto de condições de entrada que permite a possibilidade de ocorrência do primeiro evento do roteiro. O fim da cadeia é o conjunto de resultados que permitirá a ocorrência de eventos posteriores.

O raciocínio com roteiros serve especialmente para verificar se determinado evento ocorreu e também para verificar a relação entre os eventos, o que pode ser conseguido pelo exame da cadeia causal.

Barr e Feigenbaum [4], apresentam um bom exemplo de roteiro (comer-em-restaurant), aqui reproduzido e adaptado:

ROTEIRO: Comer-em-restaurant

Apoio	: (restaurante, dinheiro, alimento, menu, mesas, cadeiras)
Funções	: (pessoas com fome, encontro de pessoas)
Ponto-de-vista	: (pessoas com fome)
Tempo-de-ocorrência	: (tempo-de-operação do restaurante)
Lugar-de-ocorrência	: (localização do restaurante)

Seqüência-de-eventos

primeiro	: Inicie o roteiro entrada-no-restaurant
então	: Se (há-convite-para-sentar ou reservas) então siga roteiro orientação-do-garçom
então	: siga roteiro aguarde-sentado
então	: siga roteiro solicite-comida

então	: siga roteiro comer, a menos que haja uma longa espera,
	: caso em que seguirá o roteiro sai-do-restaurante-furioso
então	: Se (qualidade-da-comida estava melhor do que a esperada)
	então siga o roteiro cumprimentando o “chef”
então	: siga o roteiro pague-a-conta
finalmente	: siga o roteiro sai-do-restaurante

3.4.3.2.5 - Herança como Representação do Conhecimento

O mecanismo de herança tem se tornado bastante popular a medida que cresce a popularidade do paradigma objeto. Uma grande motivação é a economia de esforço para representar conhecimento em um determinado assunto.

Seja, por exemplo, representar definições da taxonomia dos animais. Pode-se partir definindo o que caracteriza o reino animal. No momento de definir um vertebrado bastará dizer que o vertebrado é um animal com coluna vertebral não sendo mais necessário dizer o que é um animal. Da mesma forma para definir o que é um mamífero basta dizer que é um vertebrado que tem glândulas secretoras de um líquido com que alimenta os filhos ao nascer e enquanto jovens.

3.4.3.2.6 - Lógica das Proposições e dos Predicados

Como todas as outras formas de representação do conhecimento, apresenta algumas deficiências. Destacam as duas seguintes:

Não tem conceitos;

Apresenta grande dificuldade na legibilidade e expressão do conhecimento por parte de pessoas não familiarizadas com a mesma, que claro está, não é um requisito exigível a um perito para que este expresse os seus conhecimentos.

3.4.3.2.6.1 - Lógica das Proposições

Proposições são afirmações que admitem um valor lógico, “verdadeiro” ou “falso”. Podemos, então, verificar a veracidade de afirmações como:

A tarde está bonita
O professor foi pescar

A cada proposição será atribuído o valor lógico verdadeiro, se as informações disponíveis permitirem tirar esta conclusão; caso contrário, será atribuído o falso.

Juntando proposições simples, através de conectivos, obteremos proposições compostas, aumentando, assim, a capacidade de expressão por meio deste recurso.

3.4.3.2.6.2 - Lógica dos Predicados

Freqüentemente a lógica dos predicados é também referida por lógica de primeira ordem ou, ainda, cálculo dos predicados.

Predicados são declarações a respeito de objetos em si ou sobre as relações dos objetos entre si.

A lógica das proposições é uma ferramenta apresentando muitas limitações, porém ela permite fazer raciocínios com facilidade.

A lógica proposicional foi expandida para a lógica dos predicados, introduzindo funções, termos, quantificadores e predicados para dar-lhes maior capacidade de expressão. Com este enriquecimento, a lógica dos predicados nos dá bons meios para raciocinar com o conhecimento, permitindo deduzir novos fatos a partir dos já existentes.

Para exemplificar a representação de informação na lógica dos predicados, considere as seguintes declarações:

- 1) Calabar foi enforcado;
- 2) Getúlio foi presidente;
- 3) Todo traidor é enforcado;
- 4) Todos os índios eram selvagens;
- 5) Tiradentes não era índio;
- 6) Tiradentes foi considerado traidor.

Sendo a representação do quantificador universal, o quantificador existencial e a negação, estas declarações têm, na lógica dos predicados, a seguinte representação:

- 1) enforcado (Calabar);
- 2) presidente (Getúlio);
- 3) $\forall x$ traidor (x) enforcado(x);
- 4) $\forall x$ índio(x) selvagem(x);

5) \neg índio(Tiradentes);

6) traidor(Tiradentes).

Uma observação rápida nos revela que as representações com predicados ocasionaram a perda de informações, como é o caso dos tempos de ocorrência dos fatos. Se este tipo de informação é importante, então incrementos na representação deverão ser feitos.

As declarações acima não são facilmente trabalháveis, pois envolvem quantificadores, “implica” e variáveis. A matemática tem uma forma para trabalhar expressões lógicas, a resolução, mas condiciona que elas estejam na forma de cláusulas.

A cláusula possui uma forma mais “linear”, a forma normal conjuntiva, sem quantificadores embutidos e sem os sinais de “implica”.

Algumas expressões, as mais simples, têm sua versão direta, para reduzir fórmulas lógicas como as acima à forma normal conjuntiva. Para as mais complicadas pode-se seguir os passos abaixo relacionados:

- 1) Eliminar as implicações. Para isto aproveita-se o fato de que $(a \rightarrow b)$ é equivalente $(\neg a \vee b)$;
- 2) Reduzir o alcance da negação, usando para isto as leis de De Morgan, a relação “ $\neg(\neg a) = a$ ” e a correspondência entre quantificadores;
- 3) Padronizar as variáveis, de forma que cada quantificador envolva apenas uma única variável. Isto é importante porque os quantificadores serão em seguida eliminados e as variáveis que aparecem em fórmulas diferentes, como o caso dos x nas fórmulas 3 e 4 acima, não serão distinguidas se isto for feito anteriormente;
- 4) Fazer uma “Skolemização”, eliminando os quantificadores existenciais. Isto é possível se substituirmos a variável “existencializada” por uma referência à função, capaz de produzir o valor desejado. As funções assim produzidas chamam-se funções de Skolem (daí o nome do processo). Funções sem argumentos são, por vezes, chamadas de constantes de Skolem;
- 5) Deslocar os quantificadores universais para o início da fórmula, sem, no entanto, modificar sua ordem relativa;

- 6) Eliminar os quantificadores universais, devendo as variáveis que ficaram, ser entendidas como universalmente quantificadas, dispensando, assim, esta simbologia;
- 7) Eliminar as conjunções. Consiste em substituir as fórmulas que contêm E's lógicos por outras logicamente equivalentes.

Efetuosos esses sete passos, teremos um conjunto de cláusulas. Cada expressão gerou apenas uma cláusula, a qual pode ser atômica ou molecular, dependendo de que tenha um ou mais literais. Assim as declarações acima dariam as seguintes cláusulas:

- 1) enforcado(Cabalar);
- 2) presidente(Getúlio);
- 3) \neg traidor(x) v enforcado(x);
- 4) \neg índio(y) v selvagem(y);
- 5) \neg índio(Tiradentes);
- 6) traidor(Tiradentes).

3.4.4 - Observações

3.4.4.1 - Representação Procedimental

A lógica dos predicados levou a uma linguagem de programação quase declarativa, o PROLOG. Nas linguagens declarativas é realçado o “o que” sobre o “como”.

Há, no entanto, a visão oposta, onde a programação detalhada é o principal veículo de representação, sendo conhecida por representação “procedimental”. Historicamente esta forma tem sido usada como suporte de retaguarda para sistemas parte declarativos e parte “procedimentais”.

Um exemplo desta forma de representação é SHRDLU, sistema apresentado por Winograd [73], o qual conversa em inglês, a respeito de um mundo de blocos.

SHRDLU explorou um pequeno corpo de conhecimento declarativo de fatos simples, como:

(É B1 BLOCO)
(ESTÁ B1 (LOCAL 100 100 O))

(MANEJÁVEL B1)
(COR-DE B1 VERMELHO)

Mas SHRDLU representou sua compreensão das declarações como um conjunto de procedimentos que fizessem o que as declarações solicitavam. Esses procedimentos foram escritos em PLANNER, Hewitt [39]. Os programas em PLANNER consistem, entre outras coisas, em declarações GOAL (meta), que descrevem estados desejados. PLANNER possui um mecanismo embutido de volta atrás, que utiliza para ajudar a encontrar um meio de satisfazer cada GOAL declarado.

Para exemplo da utilização do PLANNER pelo SHRDLU como um mecanismo de representação do conhecimento, considere a frase:

Um cubo vermelho que apóia uma pirâmide

SHRDLU teria representado esta frase como a estrutura apresentada na figura 08. A compreensão pelo SHRDLU da frase descritiva é um procedimento que localiza um objeto que satisfaça a descrição. Primeiro, um objeto que satisfaça a primeira parte da descrição.

(BLOCO) é encontrado. Então esse objeto é conferido para ver se satisfaz a parte seguinte (VERMELHO). Caso positivo, o processo continua até que todos os atributos tenham sido conferidos. Se, em algum ponto, o processo falhar, o mecanismo de volta atrás do PLANNER é invocado e um novo objeto é escolhido para investigação.

FIGURA 08 - UMA REPRESENTAÇÃO PROCEDIMENTAL DE UMA FRASE DESCRITIVA

(GOAL (É ?X1 BLOCO))
(GOAL (COR_DE ?X1 VERMELHO))
(GOAL (EQUIDIMENSIONAL ?X1))
(GOAL (É ?X2 PIRÂMIDE))
(GOAL (SUPORTA ?X1 ?X2))

Muito poucos sistemas exploram o conhecimento procedimental, com exceção das formas declarativas. Mesmo SHRDLU, utilizou algum conhecimento declarativo. Nenhum sistema pode sobreviver exclusivamente do conhecimento declarativo, sem nenhum procedimento para manipular aquilo que sabe. Em vez disso, a maioria dos sistemas utiliza uma combinação dos dois métodos. As linguagens de programação de IA, como o LISP e o KRL, facilitam esta abordagem mista ao permitirem que

procedimentos e formas declarativas sejam combinados em uma única estrutura de dados. A associação de procedimentos a estruturas de dados a que são aplicáveis é denominada procedimento de ligação e provou ser útil em muitos domínios.

Por exemplo, considere um sistema de quadro em que cada objeto é descrito como um conjunto de escaninhos, cada um preenchido com um valor. A figura 09(A) mostra uma descrição simples para uma pessoa, Joe. Podemos aumentar essa descrição declarativa com informações procedimentais, tal como calcular o valor de um escaninho se o valor for atualmente desconhecido e necessário. A figura 09(B) mostra uma descrição do escaninho IDADE (visto como um conceito próprio), que contém o escaninho PARAESTABELECER. O procedimento, que é o valor daquele escaninho, descreve um meio de calcular valores IDADE quando forem necessários.

FIGURA 09 - UM EXEMPLO DE PROCEDIMENTO DE LIGAÇÃO

(A)

(JOE (É UM PESSOA)
 (NASCIMENTO(ANO 1946)
 (MÊS NOVEMBRO)
 (DIA 31))
 (ALTURA 71)

(B)

(IDADE NIL))
 (IDADE (PARAESTABELEECER (SUBTRAIRAGORA (ANO DE NASCIMENTO))))

A representação procedimental de uma peça de informação é, essencialmente, um plano para a utilização dessa informação. Assim, construir uma boa representação procedimental é semelhante a construir qualquer outro tipo de plano. Por isso, o trabalho em representação do conhecimento procedimental está intimamente ligado ao trabalho na geração de planos. Linguagens de geração de planos de alto nível, como o PLANNER, facilitam a representação do conhecimento procedimental ao esconder os detalhes do método pelo qual os planos são realizados. (Por exemplo, a volta atrás, que era necessária nos problemas do mundo de blocos considerados pelo SHRDLU, não precisa ser representada explicitamente, pois é tratada pelo PLANNER).

Observe, porém, que à medida que linguagens de nível cada vez mais alto forem usadas para descrever o conhecimento procedimental, exigências sempre mais pesadas serão colocadas para o interpretador (ou compilador) da linguagem. Eventualmente, as próprias representações do conhecimento virão a parecer muito declarativas, enquanto todos os mecanismos para raciocinar com elas estarão contidos no interpretador. Deste argumento, deve ficar claro que não há fronteira clara entre formas declarativas de representação e as representações procedimentais. Ao contrário, há um espectro, e a escolha da posição correta, ao longo desse espectro, para a representação de uma base de conhecimento em particular depende do próprio conhecimento e da utilização para a qual ele deve ser colocado.

3.4.4.2 - Dependência Conceitual

A dependência conceitual (muitas vezes apelidada de CD) é a teoria de como representar o significado de frases em linguagem natural de modo a:

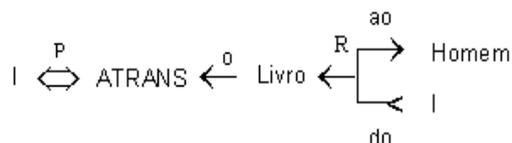
Facilitar a geração de inferências das frases;

Ser independente da linguagem em que a frase foi originalmente declarada.

Por causa dessas duas observações, a representação CD de uma frase é construída não de primitivos correspondentes às palavras utilizadas na frase, mas de primitivos conceituais que podem ser combinados para formar os significados de palavras em qualquer linguagem em particular. Essa teoria foi implementada numa variedade de programas que lêem e compreendem redações em linguagem natural.

Um exemplo simples de como o conhecimento é representado em CD é o evento representado pela frase:

Dei ao homem um livro
que seria representado conforme abaixo:



Onde os símbolos possuem os seguintes significados:

Setas indicam a direção da dependência.

Seta dupla indica um elo de mão dupla entre o ator e a ação.

P indica tempo passado.

ATRANS é um dos atos primitivos utilizados pela teoria. Ele indica transferência de posse.

I indica a relação de caso do objeto.

R indica a relação de caso do recipiente.

A dependência conceitual fornece não apenas uma estrutura onde o conhecimento pode ser representado, mas também um conjunto específico de blocos de construção dos quais as representações podem ser construídas. Um conjunto desses contém um grupo de ações primitivas, onde as de nível mais alto, correspondentes a palavras na língua, podem ser construídas. Embora haja ligeiras diferenças no conjunto exato de ações primitivas fornecidas nas várias fontes sobre CD, um conjunto típico é o seguinte, Schank [64]:

ATRANS	Transferência de uma relação abstrata (por exemplo, dar)
PTRANS	Transferência da localização física de um objeto (por exemplo, ir)
PROPEL	Aplicação de força física a um objeto (por exemplo, empurrar)
MOVE	Movimento de uma parte do corpo de alguém (por exemplo, chutar)
GRASP	Segurar um objeto por um ator (por exemplo, jogar)
INGEST	Ingestão de um objeto (por exemplo, comer)
EXPEL	Expulsão de algo do corpo de um animal (por exemplo, chorar)
MTRANS	Transferência de informação mental (por exemplo, dizer)
MBUILD	Construir nova informação da antiga (por exemplo, decidir)
SPEAK	Produção de sons (por exemplo, dizer)
ATTEND	Enfocar o órgão de sentido em um estímulo (por exemplo, ouvir), denominado CONC em alguns trabalhos anteriores de Schank [65].

Um segundo conjunto de blocos de construção é o conjunto de dependências permitidas entre as conceitualizações descritas numa frase. Há quatro categorias de conceitualizações primitivas cujas estruturas de dependência podem ser construídas. São elas:

ACTs	Ações
PPs	Objetos (produtores de imagens)
AAs	Modificadores de ações (auxiliadores de ação)
PAAs	Modificadores de PPs (auxiliadores de imagens)

Além disso, as estruturas de dependência são, elas próprias, conceitualizações e podem servir como componentes de estruturas de dependência maiores.

As dependências entre as conceitualizações correspondem a relações semânticas entre os conceitos subjacentes. A figura 10 lista algumas das mais importantes, permitidas pelo *CD*. A primeira coluna contém as regras; a segunda, contém exemplos de sua utilização; e a terceira contém uma versão, em linguagem natural, de cada exemplo. As regras apresentadas na figura podem ser interpretadas conforme segue:

Regra 1 descreve as relações entre um ator e o evento que ele causa. Esta é uma dependência de mão dupla, pois nem o ator nem o evento podem ser considerados primários. A letra *p* acima do elo de dependência indica tempo passado.

Regra 2 descreve as relações entre um PP e o PA que esta sendo afirmado como o descrevendo. Muitas descrições de estado, como altura, são representadas no *CD* como escalas numéricas.

Regra 3 descreve a relação entre dois PPs, um dos quais pertence a um conjunto definido pelo outro.

Regra 4 descreve a relação entre um PP e um atributo que já foi predicado dele. A direção da seta é no sentido do PP que está sendo descrito.

Regra 5 descreve a relação entre dois PPs, um dos quais fornece um tipo particular de informação sobre o outro. Os três tipos mais comuns de informações a serem providas desta forma são posse (apresentada como POSS-POR), localização (apresentada como LOC) e abrangência física (apresentada como CONT). A direção da seta novamente é no sentido do conceito que está sendo descrito.

Regra 6 descreve a relação entre uma ACT e o PP que é objeto daquela ACT. A direção da seta é no sentido da ACT, pois o contexto da ACT específica determina o significado da relação de objeto.

Regra 7 descreve a relação entre uma ACT e a fonte e o recipiente da ACT.

Regra 8 descreve a relação entre uma ACT e o instrumento com o qual é realizada. O instrumento deverá sempre ser uma conceitualização integral (isto é, ele deverá conter uma ACT), e não apenas um único objeto físico.

Regra 9 descreve a relação entre uma ACT e sua origem e destino físico.

Regra 10 representa a relação entre um PP e um estado em que ele começou e outro em que ele terminou.

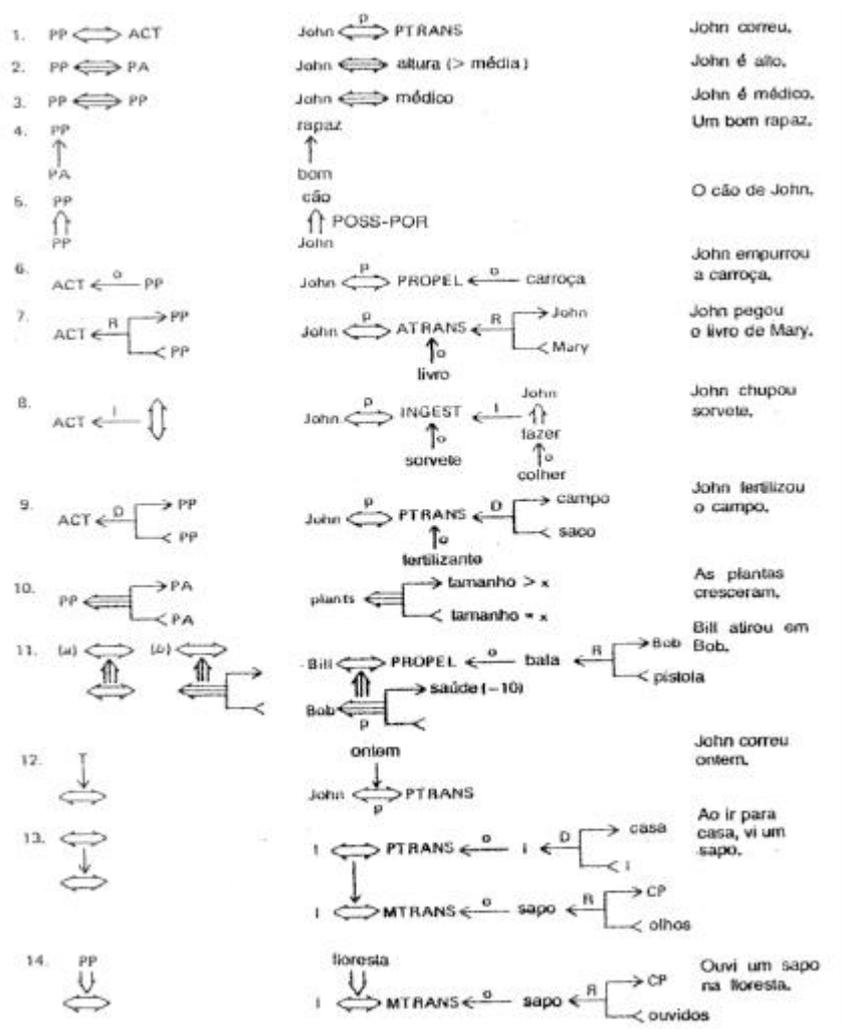
Regra 11 descreve a relação entre uma conceitualização e outra que a causou. Observe que as setas indicam dependência de uma conceitualização sobre a outra e assim apontam na direção oposta às setas de implicação. As duas formas da regra descrevem a causa de uma ação e a causa de uma mudança de estado.

Regra 12 descreve a relação entre uma conceitualização e o tempo da ocorrência do evento que ela descreve.

Regra 13 descreve a relação entre uma conceitualização e outra que é o tempo da primeira. O exemplo para esta regra também mostra como CD explora um modelo do sistema de processamento de informação humana; ver é representado como a transferência de informação entre os olhos e o processador consciente.

Regra 14 descreve a relação entre uma conceitualização e o local da sua ocorrência.

FIGURA 10 - AS DEPENDÊNCIAS DE CD



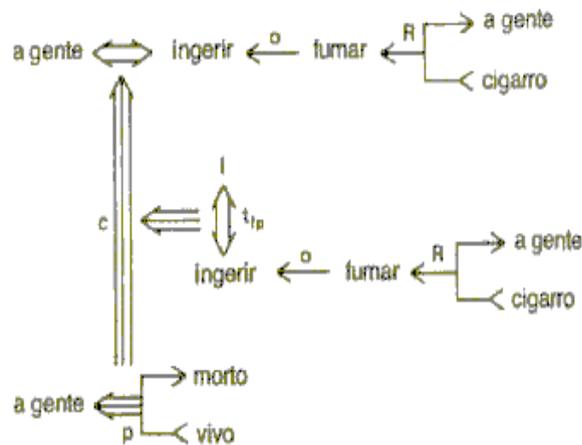
As conceitualizações que representam eventos podem ser modificadas de muitos modos para suprir informações normalmente indicadas na linguagem pelo tempo, modo ou aspecto da forma do verbo. A utilização do modificador *p* para indicar tempo passado já foi mostrada. O conjunto de tempos conceituais propostos por Schank [65] inclui:

- p Passado
- f Futuro
- t Transição
- ts Início de transição
- tf Fim de transição
- k Continuidade
- ? Interrogativo
- / Negativo
- nil Presente
- delta Intemporalidade
- c Condicional

Como um exemplo da utilização desses tempos, considere a representação CD, mostrada na figura 11 (tirada de Schank [65]), da frase:

FIGURA 11 - UTILIZAÇÃO DE TEMPOS CONCEITUAIS

Como o fumar pode lhe matar, eu parei.



O elo de causalidade vertical indica que o fumar mata a gente. Como ele está marcado com *c*, entretanto, sabemos apenas que o fumar pode matar a gente, não que ele necessariamente o faça. O elo de causalidade horizontal indica que é essa primeira causalidade que me fez parar de fumar. A qualificação *tp* afixada à dependência entre EU e INGERIR indica que o fumar (um caso de INGERIR) parou e que a parada ocorreu no passado.

3.5 - MECANISMOS DE RACIOCÍNIO

O raciocínio é a faculdade humana de chegar a conclusões a partir de conhecimentos ou informações sobre um problema.

Na continuação apresentam-se algumas formas de raciocínio:

Raciocínio com Regras de Produção

Raciocínio Probabilístico

Raciocínio Qualitativo

Raciocínio Nebuloso

Raciocínio Plausível

3.5.1 - Raciocínio com Regras de Produção

O raciocínio com regras de produção envolve o gerenciamento da(s) base(s) de regras, a manutenção do contexto em que o sistema está inserido, a interpretação adequada das regras e a aplicação de algum algoritmo de solução, que servirá de base para a estrutura do sistema “presente”, ou a máquina de inferência.

Acoplado ao algoritmo de solução deve haver recursos para optar entre as diversas formas de busca; as quais são apresentadas posteriormente.

O gerenciamento da base de regras envolve o controle do conjunto de fatos através das informações do quadro-negro (rascunho), o acréscimo e retirada de regras e a verificação de sua consistência.

O interpretador de regras age ciclicamente, sendo que em cada ciclo descobre as regras aplicáveis (através do “matching”), escolhe a regra que será aplicada, resolve conflitos e “dispara” as ações resultantes da aplicação da regra escolhida.

No raciocínio com regras de produção, dois aspectos são ressaltados: a estratégia de controle e a resolução de conflitos. A estratégia de controle é, via de regra, uma imposição relacionada com o tamanho do sistema (número de regras), enquanto que a resolução de conflitos está relacionada mais com o tipo de árvore (ou grafo) de inferência produzido.

Estratégia de Controle

As regras dos sistemas de produção estão ficando, gradativamente, mais complexas, tanto no lado esquerdo como no lado direito. O uso adicional de variáveis complica ainda mais a natureza dos sistemas de produção, a qual é, essencialmente, combinatória. Desta forma, deve ser dada especial atenção para as estratégias de controle usadas na execução das regras.

Rich [61], analisando o impacto crítico que o número de regras terá sobre o tempo necessário para a sua execução e a qualidade das respostas que delas se poderá obter,

sugere, para as estratégias de controle, a presença, no mínimo, das duas primeiras seguintes qualidades, às quais acrescentamos mais duas:

Uma boa estratégia de controle deve causar movimento. Se a estratégia de controle não for habilidosa, a execução poderá entrar em círculo vicioso;

Uma boa estratégia de controle é sistemática. Ser sistemática implica em ter capacidade para evitar a repetição de partes da solução, tanto em nível local como em nível global. Para implementar a sistematicidade, pode-se recorrer ou à busca em largura ou em profundidade. A opção entre as duas fica por conta dos objetivos a serem alcançados e em função das exigências impostas para a solução;

Uma boa estratégia de controle deve ser estável. Ser estável implica em que na busca da solução haja certa persistência, não abandonando um ramo de busca por qualquer pequena alteração de valor, evitando saltos sucessivos entre diversas subárvores de busca;

Uma boa estratégia de controle é eficiente. Ser eficiente significa proporcionar a resposta em tempo aceitável. Isto nem sempre é possível e um exemplo pode ser visto quando se analisa o problema do caixeiro-viajante, tão conhecido de todos. Se neste problema todas as alternativas de rotas forem consideradas, um computador poderá levar anos para garantir o retorno da melhor solução. Isto começa a ocorrer com um número relativamente pequeno de pontos pelos quais o viajante deverá passar.

Resolução de Conflitos

Ao terminar o processo de busca, o motor de inferência dispõe de um conjunto de regras que satisfazem à situação atual do problema, o chamado “Conjunto de Conflitos”. Caso este conjunto seja vazio, a execução é terminada, caso contrário é necessário escolher que regras serão realmente executadas e em que ordem.

Algumas formas de encarar a resolução de conflitos são:

Aplicar a primeira regra aplicável que for detectada. Naturalmente aí está envolvido o critério de busca adotado, especialmente o “backward” e o

“forward”. Subentende-se, então, uma ordenação adequada das regras a aplicar;

Aplicar a regra de maior prioridade. Isto faz supor que a cada regra seja atribuído um número, indicando o grau de prioridade inerente, sendo as regras aplicáveis, então, ordenadas segundo este valor;

Aplicar a regra mais específica. Regras específicas costumam ser mais restritivas que regras gerais. Regras envolvendo comparações de variáveis costumam ser mais gerais do que as que comparam números;

Adotar o critério LIFO (“Last In First Out”) em relação aos elementos acrescentados ao ambiente. Ou seja, promover a execução da regra que atender a informação mais recente do sistema;

Adotar uma regra arbitrária, que pode ser apontada por sorteio;

Adotar uma nova regra. Quando o sistema “entra num marasmo”, (espécie de paralisia gerada por um processo não-sensível de seleção de regras), pode-se “injetar um novo ânimo” no sistema, impondo-lhe uma nova opção, possivelmente não-viciada;

Pôr à prova as regras aplicáveis, explorando-as em paralelo e decidindo, após isto, entre as mais promissoras.

3.5.1.1 - Exemplo de Raciocínio com Regras de Produção

Para exemplificar um raciocínio com regras de produção, consideremos o conjunto de regras a seguir, a respeito de defeitos de um motor de carro:

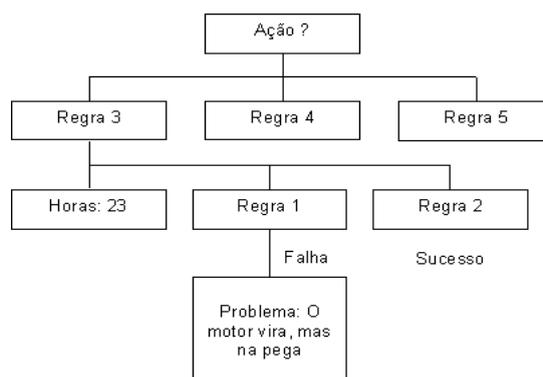
Regra 1	Se o problema for: o motor não vira, então o defeito é: arranque defeituoso.
Regra 2	Se o problema for: o motor vira, mas o motor não pega, então o defeito é: falta de combustível.
Regra 3	Se o defeito for: falta de combustível e são mais de 22 horas, então a ação será: completar o trajeto a pé.
Regra 4	Se o defeito for: arranque defeituoso e distância da oficina é maior que 6 km, então a ação será: chamar um táxi.
Regra 5	Se o defeito for: arranque defeituoso e falta de combustível, então a ação será: vender o carro.

Se uma pessoa desejar saber qual a ação que deverá ser executada com base nas regras acima, usando um engenho de inferência que executa as regras com critério “backward” e busca em profundidade, terá por primeira resposta “completar o trajeto a pé”. Esta resposta foi obtida pelo desenvolvimento da árvore de busca, conforme figura 12.

Inicialmente serão abertos (colocados na árvore) “nós” simbolizando as regras 3, 4, e 5 e dispara (expande) a regra 3.

Para disparar a regra 3, é necessário ter conhecimento das horas e saber o defeito do carro. As horas são perguntadas ao usuário, que responde 23. Então o defeito exige o disparo das regras 1 e 2.

FIGURA 12 - ÁRVORE DE BUSCA



Ao executar a regra 1, faz-se a solicitação ao usuário para informar o problema que o carro está apresentando. Recebe a informação de que “o motor vira, mas não pega”. Com esta informação, a regra 1 falha, sendo então disparada a regra 2, que apresenta sucesso.

Nesta altura o sistema pode apresentar a primeira solução (resposta), que no caso é a única, “falta de combustível”.

Em uma execução “forward”, a árvore de busca teria outra forma e as informações seriam requeridas do usuário em outra ordem.

3.5.1.2 - Técnicas de Busca

3.5.1.2.1 - Gerar e Testar

É quando se gera uma solução candidata e verifica-se se ela é realmente uma solução. É o método mais simples e freqüentemente o menos eficiente.

Supomos que desejamos encontrar as raízes de um polinômio de grau sete. O “método de gerar e testar” consiste em achar a solução, substituir a variável pelo valor achado e calcular o valor do polinômio. Se o valor é zero, a raiz candidata achada é realmente uma solução. Caso contrário não o é, e achamos outra solução candidata. Repete-se o processo até que as setes soluções são encontradas.

3.5.1.2.2 - Raciocínio para Frente Versus Raciocínio para Trás

Em um sistema especialista temos uma base de conhecimento onde encontramos fatos e regras e uma máquina de inferência que controla o uso das regras para chegar a uma conclusão.

Raciocínio direto ou para frente é começar com fatos encontrados numa base de conhecimento e manipulá-los com as regras e tentar chegar a uma conclusão. O raciocínio inverso, ou para trás, por outro lado, começa usando a conclusão e tenta provar se são verdadeiras ou falsas.

Cada caso pode ser facilmente resolvido por um método ou outro, às vezes programas que auxiliam na construção de sistemas especialistas, usualmente são denominados “shells”, permitem a escolha entre raciocínio direto e inverso e devemos saber como fazer uma boa escolha.

Uma regra prática para a escolha entre o raciocínio direto e inverso é examinar a árvore de pesquisa que será produzida em cada passo e escolher a menor.

Problemas de planejamento, projeto e classificação, tipicamente utilizam raciocínio direto, enquanto problemas de diagnósticos, onde existem apenas algumas saídas possíveis mas um grande número de estados iniciais, utilizam raciocínio inverso. Existem outros que utilizam-se do raciocínio misto, onde os raciocínios direto e inverso

se alternam de acordo com o desenvolvimento da solução de problemas e com a disponibilidade de dados.

3.5.1.2.3 - Busca em Largura

Trabalha sob um critério FIFO (First In First Out). É também denominada de busca em amplitude, busca em nível e “breadth-first”. É um procedimento em que todos os nós de certo nível da árvore são examinados antes dos do nível abaixo. Se existe uma solução e se o grafo é finito, então por este método ela certamente será encontrada. Infelizmente a busca em largura tem alguns sérios inconvenientes, tais como:

Requer muita memória. O número de nós, que é exponencial em relação à profundidade, aumenta absurdamente a quantidade de memória necessária;

Exige um esforço computacional relativamente grande, se o comprimento dos caminhos for “não-pequeno”;

Esforço despendido com operadores de pouca importância assume as mesmas proporções que o despendido com operadores mais importantes.

3.5.1.2.4 - Busca em Profundidade

Esta técnica, também denominada de primeiro-em-profundidade ou “depth-first”, explora o caminho para o objetivo, dando preferência aos nós que estão mais distantes da raiz da árvore de busca.

Nesta forma de busca o nó mais profundo em ABERTO é sempre o mais recentemente gerado. Isto permite estruturar ABERTO como uma pilha, sendo sempre selecionado para expansão o nó topo, caracterizando uma política LIFO (Last In First Out). Contudo, não prover esta pilha de algum dispositivo de controle seria perigoso, pois em certos casos a busca poderia se descontrolar, especialmente, tratando-se de grafos com profundidades infinitas ou grafos muito grandes.

Duas condições podem levar a suspender a busca e efetuar um “backtracking”, ou seja, desfazer os passos já feitos e em um certo ponto fazer escolha diferente. Primeiramente, quando o limite de profundidade é excedido, e, segundo, quando um nó é reconhecido como um impasse. Esta segunda hipótese ocorre quando um nó não

responde positivamente a algum teste que deveria satisfazer, indicando pertencer a alguma solução. Os passos do algoritmo “depth-first” são os seguintes:

- 1) Ponha o nó início em ABERTO;
- 2) Se ABERTO está vazio, então houve falha; se não, continue;
- 3) Remova o nó N do topo de ABERTO e acrescente-o em FECHADO;
- 4) Se a profundidade de N ultrapassou a profundidade limite, limpe FECHADO e volte ao passo P2; se não, continue;
- 5) Expanda N, pondo seus sucessores no topo de ABERTO, “amarrando-os” com N;
- 6) Se algum dos sucessores pertence à meta, imprima a solução;
- 7) Se algum dos sucessores for um nó impasse, retire-o de ABERTO e ajuste FECHADO;
- 8) Volte a P2.

3.5.1.2.5 - Subida da Montanha

Subida da montanha (“Hill climbing”) é algo similar ao método de otimização onde tentaremos maximizar a função que expressa a qualidade da solução candidata. Quando consideramos uma “hill climbing” podemos dar uma interpretação a uma série de soluções intermediárias. Supondo-se que encontrar a solução do problema é subir uma montanha. Para isso, podemos ir a cada ponto na direção da encosta mais alta e parar quando em qualquer direção a partir daí podemos descer. Mas este método não assegura que se atinja o ponto mais alto da montanha. Ele assegura somente que atingindo um ponto mais alto do que os vizinhos, então encontramos uma boa solução local. Uma solução global seria encontrar o ponto mais alto que pudéssemos encontrar.

3.5.1.2.6 - Abdução

Abdução funciona da seguinte maneira: Se A é a consequência de B e A é verdade conclua que B é verdade. Ora, A poderia ser consequência de um C sendo verdade que em B fosse verdade. Na realidade uma abdução poderia quanto muito indicar uma probabilidade, nunca uma inferência.

Convém ainda notar que este é o tipo de raciocínio mais comum em diagnóstico médico. Com efeito, se uma doença A implica na presença de sintoma B , um paciente apresentando o sintoma B é suspeito de estar com a doença A .

3.5.2 - Raciocínio Probabilístico

É talvez o mais antigo que trata com mecanismos de imprecisão ou incerteza. É aquele que apoiando-se em informações probabilísticas sobre fatos de um domínio chega a uma conclusão a respeito de um novo fato, conclusão está associada a uma probabilidade.

Por exemplo, quando um médico afirma: “a chance de que um paciente portador da doença D apresente no futuro próximo o sintoma S é p ”, a verdade desta afirmação não é o valor preciso de p , mas a razão específica para a crença do médico, o contexto ou a suposição sobre a qual a crença deveria estar firmemente mantida e as fontes da informação, que acarretam mudanças nesta crença.

As razões para que crenças, sendo disposições mentais sobre eventos não reprodutíveis e freqüentemente não observáveis, devam se combinar com as leis da profundidade, encontra-se examinando as relações primitivas da linguagem de probabilidades e sua possibilidade em manipular relações qualitativas.

Segundo Linares [47], [48], [49], as seguintes relações são básicas na linguagem:

Verossimilhança: “É mais verossímil que um pássaro vôle do que um peixe”. Entretanto existem peixes voadores.

Condicional: “Se um pássaro está doente, ele não pode voar”. Um pássaro doente e muitas vezes, machucado consegue voar para se escapar de um gato faminto.

Relevância: “No dia 31 de dezembro a noite se vai a alguma manifestação festiva”. A menos que se esteja numa cidade sozinho, sem conhecer ninguém.

Causal: “Se alguém está cansado, vai dormir”. A menos que exista uma motivação importante para estar acordado.

Verossimilhança

Relações qualitativas da forma “*A é mais verossímil que B*” foram um dos primeiros propósitos na criação da teoria das probabilidades em que jogadores tais como Cardano [1501-1576] e De Moivre [1667-1754] estavam interessados em saber se das duas jogadas possíveis *A* e *B* qual a que deveria ser feita por ser mais verossímil.

Condicional

A teoria da probabilidade adota a frase epistêmica “... posto que *C* é conhecido” como uma primitiva da linguagem. Sintaticamente, isto é denotado como:

$P(A | C) = p$
onde *A* é uma dada proposição.

Esta frase combina as noções de conhecimento e crença pela atribuição à *A* de um grau de crença *p*, dado o conhecimento de *C*. *C* é chamado também o contexto da crença em *A*, e a notação $P(A | C)$ é chamada probabilidade condicional de Bayes. Thomas Bayes [1702-1761] fez sua principal contribuição à ciência de probabilidade pela associação da frase “... dado que se conhece *C*” com a famosa fórmula:

$$P(A | C) = \frac{P(A, C)}{P(C)}$$

A qual chegou a ser uma definição de probabilidade condicional.

É pela probabilidade condicional de Bayes que a teoria da probabilidade facilita o raciocínio não-monotônico, isto é, o raciocínio probalilístico que envolve reavaliação de conclusões prévias. Por exemplo, é bastante aceitável afirmar simultaneamente. A crença na proposição:

“ $P(\text{voar}(a) | \text{pássaro}(a))$ é grande”, logo
 $P(\text{voar}(a) | \text{pássaro}(a)) = 1$ ou poderia estimar-se 0.99

e a crença na proposição:

“ $P(\text{voar}(a) | \text{pássaro}(a), \text{doente}(a))$ é baixa”, logo
 $P(\text{voar}(a) | \text{pássaro}(a), \text{doente}(a)) = 0$ ou poderia estimar-se 0.02

Em outras palavras, sabe-se que o objeto *a* é um pássaro, pode-se chegar à conclusão que *a* provavelmente voa. No entanto, sob o conhecimento que *a* está doente,

há uma reavaliação sobre a primeira conclusão e afirma-se que é menos provável que a possa voar.

Para facilitar tais reavaliações são necessárias duas coisas: que a crença original seja estabelecida com uma certeza menos absoluta, e que o contexto sobre a condição de crença seja consultada constantemente, para ver se na reavaliação a crença é a mesma. A dinâmica da revisão da crença sob mudanças no contexto não é totalmente arbitrária, mas deve obedecer algumas leis básicas de plausibilidade as quais, estão embutidas nas regras sintáticas do cálculo de probabilidades. Um exemplo de tais leis plausíveis é a seguinte regra:

Regra do meio hipotético: “Se duas suposições diametralmente opostas tem dois diferentes graus de crença sobre uma proposição Q , logo o grau incondicional do mérito de crença para Q seria algum valor entre os dois”.

Por exemplo, a crença que $B = \text{“Piolin voa”}$ dado que $C = \text{“Piolin é um pássaro”}$ pode estar entre a crença de que Piolin voa dado $A = \text{“Pássaro saudável”}$ e a crença que Piolin voa dado $\neg A = \text{“Pássaro doente”}$. Esta regra, restrição do senso comum, é construída dentro da sintaxe do cálculo de probabilidade através da seguinte igualdade:

$P(B | C) = \hat{a} P(B | C, A) + (1 - \hat{a}) P(B | C, \neg A)$
 onde: $\hat{a} = P(A | C)$ é algum número entre 0 e 1.

Relevância

Relevância é uma relação indicando uma mudança potencial de crença devido a uma mudança específica no conhecimento. Duas proposições A e B são chamadas relevantes no contexto C , se a inclusão de B em C pode mudar a verossimilhança de A . Relevância pode ser definida em termos de verossimilhança e condicional. Por exemplo, uma pessoa poderia estar indecisa em avaliar a verossimilhança de dois eventos, mas pode ter confiança para opinar se os eventos são ou não relevantes uns aos outros. Relevância é também uma primitiva da linguagem da teoria de probabilidade, porque a linguagem permite especificar relações relevantes direta e qualitativamente antes de fazer qualquer avaliação numérica. Depois, quando a avaliação numérica de verossimilhança é requerida, ela pode ser adicionada de forma consistente, sem atrapalhar a relevância da estrutura original, através de probabilidade condicional.

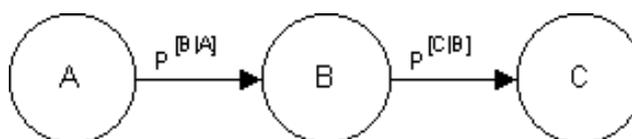
Causal

Causalidade é uma noção ubíqua na concepção do homem sobre seu meio ambiente, ainda que tradicionalmente tenha sido considerada uma construção psicológica, externa ao âmbito da probabilidade ou até das ciências físicas [56].

Causalidade é uma linguagem com a qual uma pessoa pode falar eficientemente sobre estruturas de relações relevantes, com o objetivo de separar o relevante do supérfluo. Por exemplo, dizer que o chão úmido foi uma causa direta do escorregamento e fratura da perna de uma pessoa é um caminho conciso para identificar que eventos deveriam ser considerados relevantes ao acidente, assim que a umidade do pavimento seja confirmada. O fato que choveu esse dia e que uma outra pessoa também escorregou e fraturou sua perna não devem ser considerados relevantes ao acidente, desde que se estabeleceu a verdade do “chão úmido” e se identificou como uma causa direta do acidente.

O estabelecimento de uma relação causal segue padrões de dependência: Dois eventos não chegam a ser relevantes entre si somente pelo fato de predizerem uma consequência comum, mas eles chegam a ser relevantes quando a consequência é realmente observada. Resumindo, pode-se afirmar que dois eventos com relação de causalidade são relevantes, mas nem todos os eventos relevantes tem causalidade.

FIGURA 13 - GRAFOS DIRECIONADOS



O raciocínio probabilístico é útil para modelar sistemas onde há a presença de incerteza, isto é, a ocorrência de um conjunto de evidências E leva a um vetor (V) de probabilidades das hipóteses concorrentes. Cada elemento de V representa a probabilidade condicional $P(H_i | E)$. A aquisição de conhecimento do especialista deve permitir chegar-se a uma base de conhecimentos representada como uma rede Bayesiana, contendo informações verossímeis, relevantes e com relações causais (figura 13).

As relações entre A, B e C são representadas utilizando grafos direcionados, onde aos arcos de ligação estão associadas probabilidades condicionais. Tem-se tentado solucionar os problemas de propagação de incerteza nas redes Bayesianas, através da máxima entropia.

3.5.3 - Raciocínio Qualitativo

É aquele que baseia-se em informações qualitativas para chegar a uma conclusão. Este raciocínio envolve dois pontos principais: os valores que podem ser adotados pelas variáveis no espaço quantitativo, e o tempo qualitativo, ou tempo durante o qual os valores qualitativos são válidos[7].

Na literatura existem muitos termos que expressam o conceito do Raciocínio Qualitativo. Encontram-se termos como “Senso Comum”, “Física Qualitativa”, “Física Intuitiva”, apresentados abaixo. Todos estes termos fornecem desde suas próprias abordagens o conceito do Raciocínio Qualitativo [15]. Apresentaremos sucintamente cada um deles.

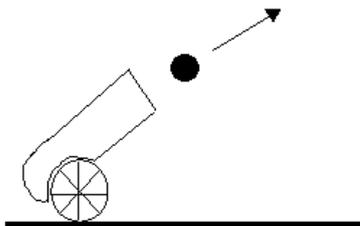
Senso Comum

Algumas pessoas acreditam que o senso comum advém de nosso armazenamento massivo de conhecimento e de nossa capacidade de acessar rapidamente os conhecimentos relevantes no momento certo. Por exemplo, quando escutamos “frente fria”, pensamos em ventos fortes, chuva e outras características a ela associadas. Acessamos nossa memória pelo conteúdo.

Física Qualitativa

Tem por objetivo saber criar e raciocinar com representações abstratas e sem números, por exemplo (ver figura 14):

FIGURA 14 - FÍSICA QUALITATIVA



Na figura 14, o modelo quantitativo que representa o fenômeno, poderia ser o seguinte: Se a velocidade inicial da bola é V_0 e o ângulo de desvio do solo for θ , então a posição da bola t segundos depois de ter sido lançada será fornecida por:

$$\begin{aligned} \text{altura} &= V_0 \cdot \text{sen}(\theta) - \frac{1}{2}g \cdot t^2 \\ \text{distância} &= V_0 \cdot t \cdot \text{cos}(\theta) \end{aligned}$$

Mas, muitas pessoas não conhecem essas equações, apesar de serem perfeitamente capazes de prever o que acontecerá. Isto é, as pessoas não precisam de medidas numéricas exatas. Elas geralmente, só precisam de descrições qualitativas. As pessoas parecem raciocinar mais abstratamente do que indicam as equações.

A física qualitativa procura compreender os processos físicos através da criação de modelos. Um modelo é uma representação abstrata que elimina os detalhes irrelevantes. Por exemplo, para prever a queda de um objeto, deve-se considerar sua massa e não sua forma ou cor. Os modelos da física tradicional são criados a partir de variáveis com valores reais, coeficientes de mudança, expressões, equações e estados. A física qualitativa oferece ferramentas semelhantes, que são mais abstratas e não-numéricas.

Física Intuitiva

É uma metodologia que tenta dirigir a questão de como o homem modela o comportamento de sistemas físicos na ausência do computador, isto é, modelar o processo de entendimento em si mesmo.

Representação de Informações Qualitativas

Variáveis - Na física tradicional, variáveis com valor real são usadas para representar características dos objetos, como posição, velocidade, ângulo e temperatura. As variáveis qualitativas adotam um conjunto ordenado finito de valores qualitativos.

Por exemplo, a quantidade de água em um pote pode ser representada por vazio, meio-termo, cheio e a sua temperatura como congelada, meio-termo, fervente.

Espaços de Quantidade - São pequenos conjuntos de valores discretos para uma variável. Os elementos de um espaço quantitativo são normalmente ordenados com base uns nos outros para que um valor possa ser considerado menor que o outro. Por exemplo a idade (figura 15).

Coefficientes de Mudança - São considerados quando as variáveis assumem valores diferentes em momentos diferentes. Um coeficiente de mudança com valor real (dx/dt) pode ser modelado qualitativamente {diminuindo, estável, aumentando}. Por exemplo a chuva (figura 16).

FIGURA 15 - ESPAÇOS QUANTITATIVOS

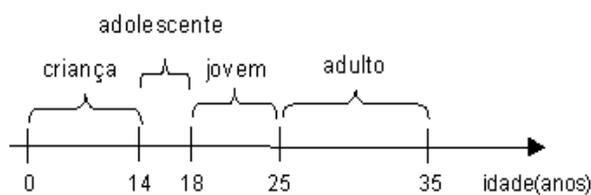
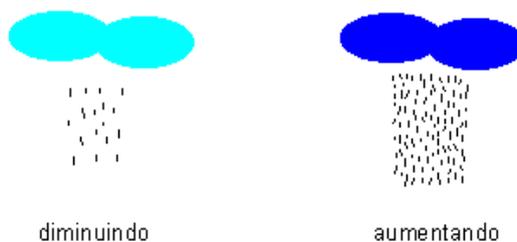


FIGURA 16 - COEFICIENTES DE MUDANÇA



Expressões - São formadas a partir da combinação das variáveis. Por exemplo: Duas variáveis qualitativas x e y , podem ser facilmente somadas. A tabela da verdade para somar variáveis qualitativas é mostrada na figura 17.

FIGURA 17 - OPERAÇÃO COM VARIÁVEIS QUALITATIVAS

	Y X	-	0	+
-		-	-	?
SOMA = 0		-	0	+
+		?	+	+

Pode-se também definir operadores qualitativos de diferença e produto. Por exemplo, considere representar o volume da água em um copo como vazio, meio-termo, cheio. Se derrarmos o conteúdo de um copo em outro, que quantidade de água o segundo copo conterá?. Pode-se somar qualitativamente do seguinte modo:

- vazio + vazio = vazio
- vazio + meio-termo = meio-termo
- vazio + cheio = cheio
- meio-termo + meio-termo = cheio? ou meio-termo?
- meio-termo + cheio = cheio? ou transbordar?
- cheio + cheio = cheio + transbordar

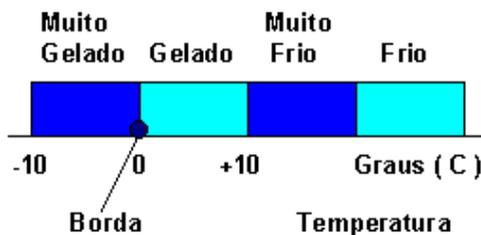
Note que a combinação de valores qualitativos pode provocar ambigüidades.

Equações - São formadas a partir de expressões e variáveis que podem ser ligadas entre si. A equação mais simples afirma que a variável y aumenta conforme aumenta a variável z. Isto nos dá uma representação abstrata da verdadeira função que relaciona x e y, por exemplo:

$$y = M^+(x)$$

Valores Limites - Seja os pontos que separam uma região de outra. Por exemplo, tem-se duas regiões + e -, o 0 viria ser um valor limite, posto que separa a região + da região -, figura (18).

FIGURA 18 - VALORES LIMITES ENTRE DUAS REGIÕES ADJACENTES



Comportamento Qualitativo - Denota um conjunto de valores ordenado no tempo de uma variável qualitativa, i.e., um episódio.

Estados Qualitativos - Um estado é um único instantâneo, onde cada variável possui um valor. Há várias maneiras diferentes de formular informações sobre estados, por exemplo: o estado da figura 16 pode ser apresentado por “Chuva caindo”.

Modelos Qualitativos - São modelos que operam sobre estados qualitativos.

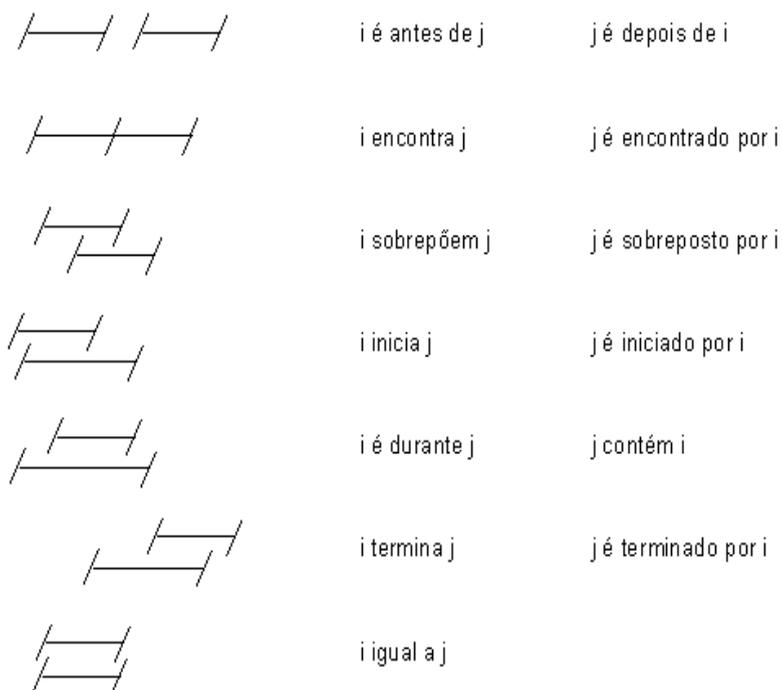
Simulação Qualitativa - É um gerador de episódio que infere comportamentos qualitativos desde um modelo qualitativo.

3.5.3.1 - Ontologias do Raciocínio Qualitativo

Tempo

A noção mais básica de tempo é que ele é ocupado por eventos. Esses eventos ocorrem durante intervalos contínuos de tempo. Os intervalos podem estar relacionados a outros intervalos, existem treze maneiras como dois intervalos de tempo não vazios podem estar relacionados entre si. Na seguinte figura mostra-se esses relacionamentos, na verdade são apenas sete relacionamentos distintos, mas seis deles têm seus próprios inversos [62] (figura 19).

FIGURA 19 - TREZE RELACIONAMENTOS POSSÍVEIS ENTRE DOIS INTERVALOS DE TEMPO

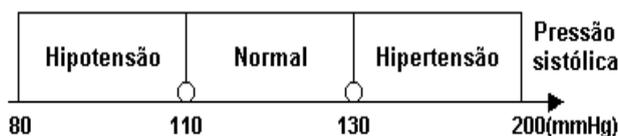


Espaço

Os objetos têm extensão espacial ao passo que os eventos têm extensão temporal. Podemos então tentar expandir a teoria de tempo para uma teoria de espaço. Mas como o espaço é tridimensional, há muito mais do que treze relacionamentos espaciais entre dois objetos, dentre elas temos: “Dentro”, “Adjacente”, “Perpendicular”, “Ao-Longo”, etc.

Pode-se ver que a representação do conhecimento no raciocínio qualitativo não é tão simples. Muitos trabalhos foram desenvolvidos nesta área, mas ainda há muito para ser feito.

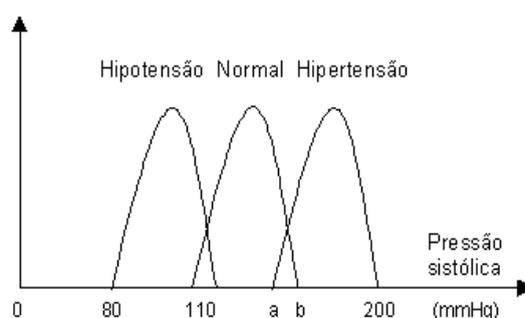
FIGURA 20 - REPRESENTAÇÃO QUALITATIVA DA PRESSÃO SISTÓLICA



O raciocínio qualitativo é útil para modelar sistemas dos quais não se tem informação completa. A forma de representação da informação qualitativa não é apropriada para ser aplicada às informações médicas em sistemas de diagnóstico clínico.

Dado que, a principal dificuldade é que ela apresenta mudanças abruptas, fato que não acontece em medicina. Por exemplo, na figura 20, observa-se que uma pequena mudança na pressão sistólica pode levar a uma interpretação de normal a patológica, fato que nem sempre é verdade, e que a lógica nebulosa pode representar melhor (figura 21).

FIGURA 21 - FUNÇÕES DE PERTINÊNCIA PARA A PRESSÃO SISTÓLICA



Neste caso, os conjuntos nebulosos permitem uma transição menos abrupta e representam melhor os conceitos clínicos. Por exemplo, tem-se uma faixa de valores [a,b] onde uma pressão sistólica pertence ao conjunto nebuloso normal e ao conjunto nebuloso hipertensão com diferentes graus de pertinência. Observa-se que a figura 21 pode ser considerada como funções de pertinência, onde os graus de pertinência são 1 (pertinência total) ou 0 (não pertinência), e esta é uma das razões pelas quais considera-se o raciocínio qualitativo como um caso particular do raciocínio nebuloso.

3.5.4 - Raciocínio Nebuloso

O raciocínio nebuloso pode-se dizer que se originou nas discussões dos gregos antigos sobre valores de verdade. Sócrates com suas perguntas a seus discípulos formulou o problema de números de verdade sem dar resposta. A corrente platônica afirmava que existiam dois valores de verdade. Mas estas discussões evoluíram até que Aristóteles, aluno de Platão, iniciou a lógica modal com quatro valores de verdade, a qual permaneceu quase no esquecimento até nosso século. As discussões filosóficas sobre o axioma do terceiro excluído (excluded middle), onde um elemento pode ser verdadeiro ou falso, e não pode ter uma terceira possibilidade (não pode ser verdadeiro e falso ao mesmo tempo), geraram paradoxos, por exemplo: “o paradoxo do barbeiro:

Em uma pequena cidade onde só existia um barbeiro. O barbeiro recebeu a missão de barbear todos os homens que não se barbeavam a si mesmos. Todos os homens deveriam estar barbeados senão morreriam. Pergunta: Quem barbearia o barbeiro?" [9].

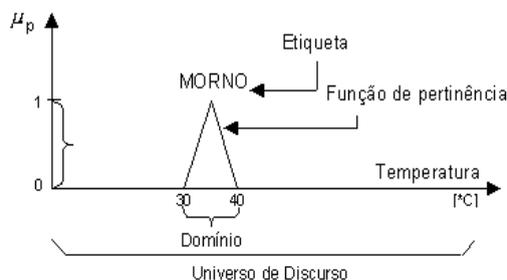
O raciocínio nebuloso foi formalizado após a publicação do artigo de Zadeh [76] nos anos 60, quando criou a Teoria de Conjuntos Nebulosos. Esta idéia foi resultado de conversas entre Zadeh e Bellman durante um ano sabático Zadeh. Eles discutiram a diferença entre os conceitos de imprecisão e incerteza, ambos tratados na época pela mesma ferramenta.

Em particular, a imprecisão (fuzziness) intrínseca da linguagem natural é uma consequência lógica da necessidade de expressar informação em forma resumida [75]. Segundo Dubois e Prade, aproximar é equivalente a raciocínio nebuloso e representa o processo ou processos pelos quais uma possível conclusão imprecisa é deduzida desde uma coleção de premissas imprecisas [24]. Tal raciocínio é, na maior parte, de natureza qualitativa mais do que quantitativa e quase tudo isto cai fora do domínio da aplicabilidade da lógica clássica.

O raciocínio nebuloso é um raciocínio que, baseado em dados imprecisos, os quais são representados por graus de pertinência a um conjunto nebuloso, chega a uma conclusão.

No raciocínio nebuloso trabalha-se com imprecisão, envolvendo processos tais como: nebulização (fuzzification), denebulização (defuzzification). E para uma melhor compreensão deles precisa-se da definição de alguns conceitos que serão apresentados na continuação (figura 22):

FIGURA 22 - FUNÇÃO DE PERTINÊNCIA



Função de pertinência: define um conjunto nebuloso mapeando entradas abruptas (crisp) de seu domínio a graus de pertinência.

Grau de pertinência: grau ao qual um valor abrupto é compatível com uma função de pertinência, o qual pode tomar valores dentro de um intervalo pré-determinado, por exemplo: [0, 1].

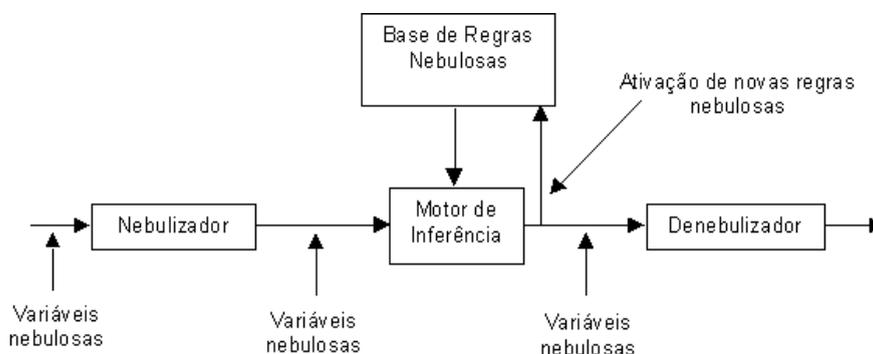
Etiqueta: nome descritivo, utiliza-se para identificar uma função de pertinência.

Domínio: valores (geralmente números) sob os quais é definida a função de pertinência.

Universo de discurso: todos os valores possíveis de uma variável do sistema.

Para explanar o caminho do raciocínio nebuloso suponha-se o seguinte sistema figura 23. Este sistema mapeia entradas abruptas em saídas abruptas e é basicamente formado por quatro componentes:

FIGURA 23 - SISTEMA DE INFERÊNCIA NEBULOSA



Regras

Podem ser fornecidas pelo especialista ou ser extraída de dados numéricos. Em ambos casos as regras de inferência são expressas como declarações do tipo “SE... ENTÃO”. Por exemplo: “SE dor na parte baixa e movimento da espinha lombar é limitado e expansão torácica diminuída e paciente mulher e idade entre 20 e 40 anos ENTÃO “espondilite anquilosante” [1]. Esta regra mostra que precisa-se de uma compreensão de:

Variáveis lingüísticas versus valores numéricos das variáveis (idade versus 35 anos);

Variáveis lingüísticas quantificadoras (dor pode ter um número finito de termos lingüísticos relacionados com ela, eles podem ir desde extremamente forte a quase nada) o qual é feito utilizando funções de pertinência nebulosa;

Conexões lógicas para variáveis lingüísticas, por exemplo, “e” “ou”, etc;

Implicações, por exemplo “Se A então B”.

E além de tudo isto, precisa-se entender como combinar mais de uma regra.

Nebulizador (Fuzzifier)

Mapeia números abrupos de entrada em conjuntos nebulosos. Isto é necessário para ativar regras que estão em termos de variáveis lingüísticas, as quais possuem conjuntos nebulosos associados a eles. Estes conjuntos nebulosos podem ser representados por funções do tipo (a) impulso ou (b) triangular ou (c) trapezoidal ou (d) S ou (e) Z ou (f) II. A função do tipo impulso pode nem sempre ser a mais apropriada, especialmente quando os dados tem uma faixa de valores que também devem ser considerados. As outras funções fornecem maior representatividade da imprecisão para serem manipuladas dentro dos *frameworks* de sistemas de lógica nebulosa.

Motor de Inferência

Mapeia conjuntos nebulosos em conjuntos nebulosos. Este manipula o caminho no qual as regras são combinadas. Da mesma maneira como os humanos utilizam muitos tipos diferentes de procedimentos inferenciais na compreensão de pensamentos ou para a tomada de decisões, existem diversos processos inferenciais em lógica nebulosa, por exemplo tem-se o sistema para estacionar corretamente um caminhão de retrocesso na doca [51] (figura 25):

FIGURA 24 - FORMAS DE REPRESENTAÇÃO DE CONJUNTOS NEBULOSOS

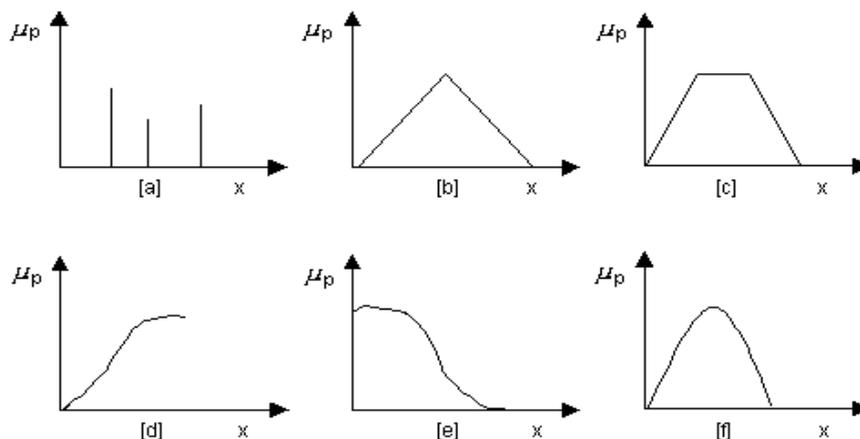
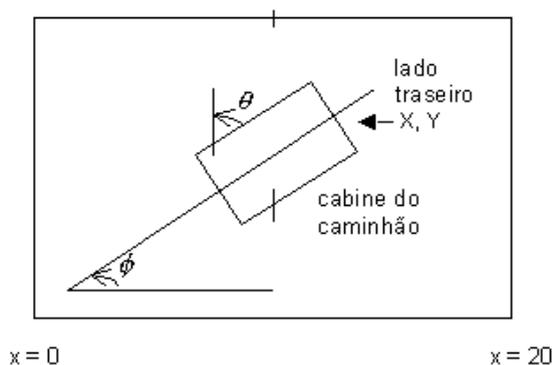


FIGURA 25 - POSIÇÃO DA CABINA DO CAMINHÃO EM RELAÇÃO A DOCA

Posição ótima para estacionar
 $X = 10$ e $\phi = 90^\circ$



Quando $x = 10$ e $\phi = 90^\circ$, o valor de q é o ótimo para estacionar o caminhão, as funções de pertinência para as duas variáveis (figura 26), onde S é pequeno, CE é centrado, B é grande.

As regras para este sistema são mostrados na tabela 01.

Suponha-se que em T_1 $x = 6$ e $\phi = 140^\circ$, observa-se nas funções de pertinência que se ativam dois conjuntos nebulosos, o qual ativa três regras (ver na tabela 3.1):

$R^{5.1}$: Se f é B1 e x é S2 logo q é B2

$R^{5.2}$: Se f é B1 e x é S1 logo q é B3

FIGURA 26 - FUNÇÕES DE PERTINÊNCIA PARA O EXEMPLO DO CAMINHÃO

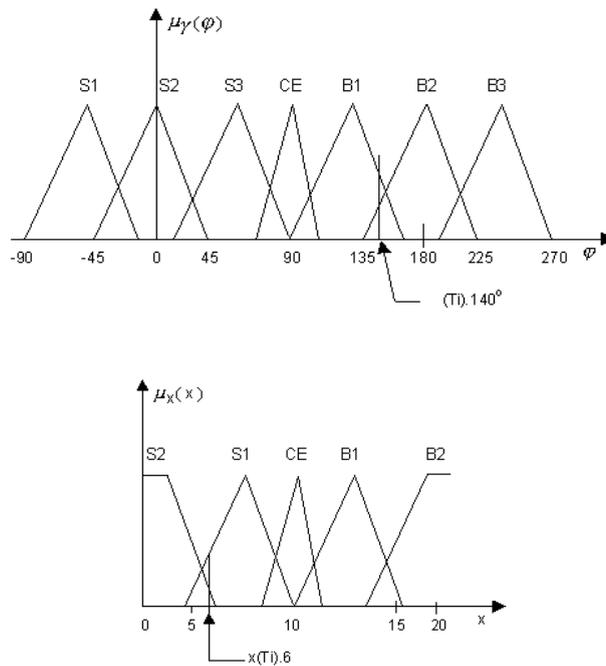


TABELA 02 - REGRAS PARA O EXEMPLO DO CAMINHÃO

$f \setminus X$	S2	S1	CE	B1	B2
S3	S2	S3			
S2	S2	S3	S3	S3	
S1	B1	S1	S2	S3	S2
C3	B2	B2	CE	S2	S2
B1	B2	B3	B2	B1	S1
B2		B3	B3	B3	B2
B3				B3	B2

$R^{6,2}$: Se f é B2 e x é S1 logo q é B3

A ativação das regras $R^{5,1}$, $R^{5,2}$, $R^{6,2}$ são mostradas na figura 27, 28 e 29 respectivamente.

Os conjuntos nebulosos obtidos usando t-conorma são mostrados na figura 30. Observa-se que as funções de pertinência de saída da inferência min apresentam um truncamento no valor de saída nebulosa, este truncamento chama-se λ -corte (λ -cut), a diferença do limiar a -corte (a -cut) que é estabelecida pelo projetista, o λ -corte (figura 31(a)) é determinado pela máquina de inferência nebulosa durante a execução de seu processo. O propósito mais comum do a -corte (figura 31(b)) é acelerar o processo de inferência.

FIGURA 27 - ATIVAÇÃO DA REGRA R^{5,1} E AS SUAS SAÍDAS

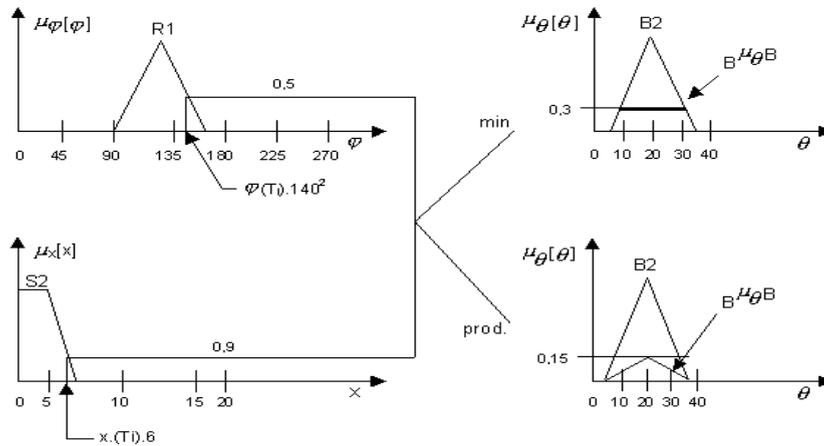
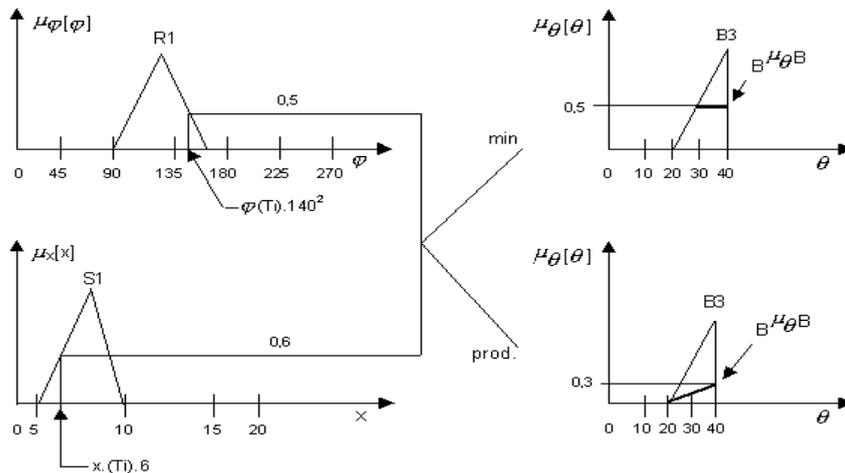


FIGURA 28 - ATIVAÇÃO DA REGRA R^{5,2} E AS SUAS SAÍDAS



Denebulizador (Defuzzifier)

Mapeia conjuntos nebulosos de saída em números abruptos. Em uma aplicação de controle, por exemplo, um número corresponde a uma ação de controle, em uma aplicação de processamento de sinais, poderia corresponder a uma predição do comportamento do sinal observado, ou em uma aplicação de diagnóstico médico a um prognóstico de doença.

Muitos denebulizadores foram propostos na literatura, embora, não existam bases científicas para nenhum deles [51] em conseqüência, denebulizar é uma arte mais do que uma ciência, dentre elas tem-se:

Denebulizador máximo. Este denebulizador examina o conjunto nebuloso B e escolhe como sua saída o valor de y para o qual : $\mu_B(y)$ é o máximo. Isto pode direcionar a resultados peculiares ou pode ter valores altos sem levar em consideração todos os valores sobre os quais encontra-se representado o conjunto nebuloso.

FIGURA 29 - ATIVAÇÃO DA REGRA R^{6,2} E AS SUAS SAÍDAS

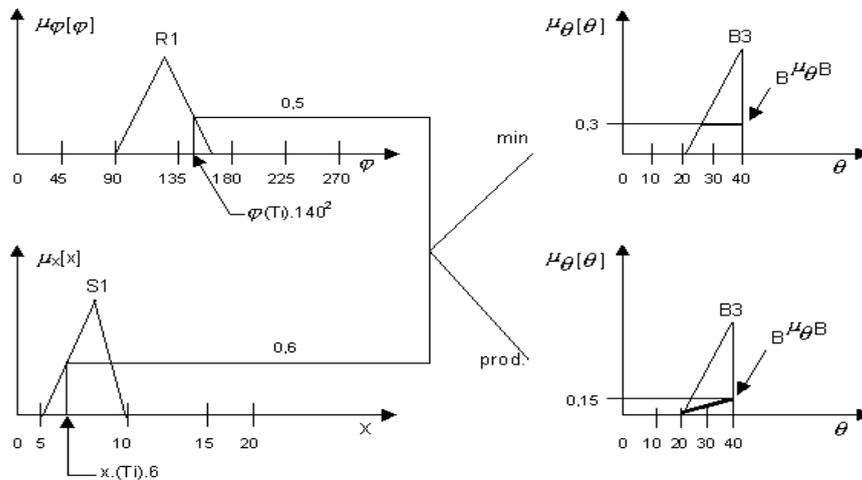
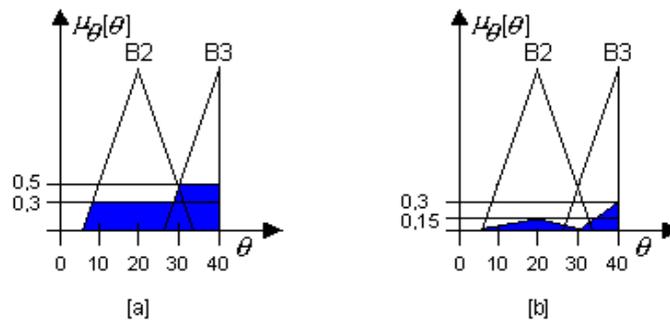


FIGURA 30 - OS CONJUNTOS NEBULOSOS OBTIDOS UTILIZANDO O T-CONORMA PARA AS INFERÊNCIAS: (A) MÍNIMO E (B) PRODUTO



Denebulizador meio do máximo. Este examina o conjunto nebuloso B e primeiro determina o valor de y para o qual $\mu_B(y)$ é o máximo. Logo computa o meio deste valor como sua saída. Infelizmente, isto pode levar a resultados inesperados, suponha-se que B está representado por dois conjuntos nebulosos com o mesmo pico de amplitude, este método obterá um valor na metade do caminho entre os dois conjuntos, onde a função de pertinência B tem o valor zero (figura 32).

Denebulizador do centróide. Este denebulizador determina o centro de gravidade (centróide), ζ de B e utiliza este valor como a saída do sistema, calculado por:

$$\tilde{y} = \frac{\int_S y \mu_B(y) dy}{\int \mu_B(y) dy}$$

FIGURA 31 - (A) λ -CORTE E (B) α -CORTE EM CONJUNTOS NEBULOSOS

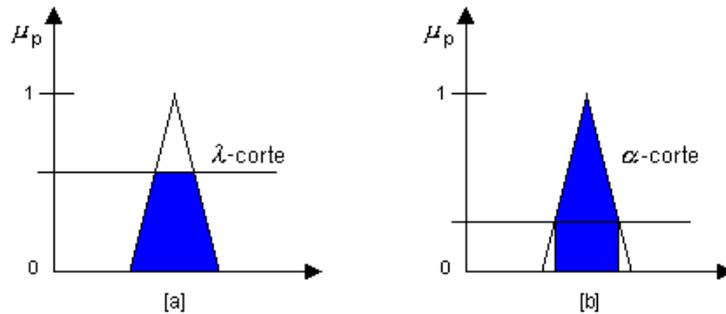
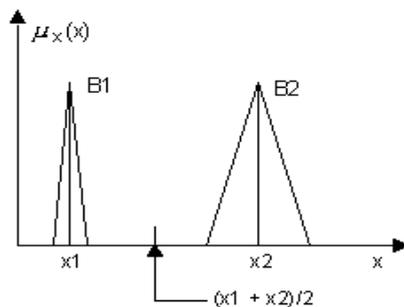


FIGURA 32 - EXEMPLO PARA O QUAL O DENEBUGADOR MEIO DO MÁXIMO NÃO FAZ SENTIDO



Onde S denota o domínio de $\mu_B(y)$. Frequentemente S é discretizado, de modo que ζ pode ser aproximado pela seguinte expressão, a qual utiliza o somatório no lugar da integração:

O denebulizador do centróide é único, no entanto, este é usualmente difícil de calcular.

$$\tilde{y} = \frac{\sum_{i=1}^I y_i \mu_B(y_i)}{\sum_{i=1}^I \mu_B(y_i)}$$

3.5.4.1 - Lógica Nebulosa e Probabilidade

Quando se apresenta pela primeira vez a noção de conjuntos nebulosos, as pessoas têm a impressão que é um disfarce da probabilidade, ou pior ainda, que tenta resolver o mesmo tipo de problema. Uma razão poderia ser, que os valores de pertinência e probabilidade se encontram em um intervalo fechado $[0,1]$.

Existe uma diferença fundamental entre probabilidade e lógica nebulosa a qual, uma vez melhor compreendida, levará a uma cuidadosa diferenciação entre as caracterizações de graus de pertinência versus graus de probabilidade, em especial em conversações formais como, diagnóstico médico, e em geral, nessas aplicações nas quais o alto grau de precisão do significado é importante alcançar.

Como os dois conceitos são muito diferentes, serão apresentados duas diferenças: semânticas e axiomáticas de cada conceito [8].

Diferenças Semânticas

A primeira diferença está no significado da pertinência nebulosa e o valor da probabilidade. No conjunto nebuloso, um valor de pertinência entre 0 e 1 significa que o elemento não pertence perfeitamente ao conjunto considerado. Por exemplo, um homem com uma temperatura de 38 graus tem febre, mas ter 40 graus é ter mais febre. Na terminologia dos conjuntos nebulosos o primeiro homem tem um grau de pertinência ao conjunto de pertinência FEBRE menor do que o segundo homem. Ambos pertencem ao conjunto nebuloso (FEBRE), só que com diferente grau de pertinência.

Continuando com o exemplo de febre, um homem que tem uma infecção provavelmente tem febre porque a infecção causa febre. Nós podemos estabelecer a probabilidade na ocorrência de febre dado que sabemos que o homem tem infecção.

O exemplo anterior mostra a importância na distinção dos conceitos nebuloso e probabilidade, ambos úteis em aplicações biomédicas. Ambos os conceitos podem ser utilizados conjuntamente.

Diferenças Axiomáticas

As diferenças axiomáticas são mais abstratas, mas convincentes. Informalmente, poderíamos dizer que em ambos os casos, pertinência nebulosa e probabilidade, temos uma função com uma gama de valores no intervalo $[0,1]$. Mas em conjuntos nebulosos o domínio da função é o conjunto universal, e no caso de probabilidades o domínio é um conjunto de partes do universo.

3.5.5 - *Raciocínio Plausível*

É a maneira de chegar a uma conclusão a partir do conhecimento de evidências sobre o fato que se está analisando.

Estritamente falando, todos os nossos conhecimentos externos à matemática e à lógica demonstrativa, consistem de conjecturas. Existem, conjecturas e conjecturas. Há conjecturas respeitáveis e formais como as expressas em certas leis da ciência física e há outras conjecturas que não são formais, nem respeitáveis, porém elas poderiam ser aceitas ou não. E assim por diante tem-se toda sorte de conjecturas, pressentimentos e suposições [57].

Os conhecimentos matemáticos são assegurados pelo raciocínio demonstrativo, mas nossas conjecturas são mantidas pelo raciocínio plausível. A comprovação

Matemática é raciocínio demonstrativo, mas as evidências não o são. Por exemplo, a evidência indutiva do físico, a evidência circunstancial do advogado, a evidência documentaria do historiador, a evidência de sinais e sintomas do médico pertencem ao raciocínio plausível.

Há diferenças entre os dois tipos de raciocínio. O raciocínio demonstrativo é seguro, longe de controvérsias e final. O raciocínio plausível é arriscado, com controvérsias e transitório. O raciocínio demonstrativo está nas ciências da mesma forma como a matemática, em si mesmo, isto é, não produz em essência novos conhecimentos do mundo real. Alguma coisa nova que se aprende sobre o mundo envolve raciocínio plausível, sendo assim é o tipo de raciocínio pelo qual se tem interesse no dia a dia. O raciocínio demonstrativo tem padrões rígidos, codificados e tratados pela lógica demonstrativa, a qual é a teoria do raciocínio demonstrativo. Os

padrões do raciocínio plausível são claros, e não há teoria de tal raciocínio que possa ser comparado à lógica demonstrativa, em clareza e que possa ter consenso.

Os raciocínios demonstrativo e plausível não contradizem um ao outro; pelo contrário se completam. No raciocínio demonstrativo o assunto principal é distinguir uma demonstração a partir de uma suposição, uma demonstração válida a partir de uma tentativa inválida. No raciocínio plausível o objetivo principal é distinguir uma suposição de outra, uma suposição mais razoável de uma suposição menos razoável.

O uso eficiente do raciocínio plausível é uma habilidade prática e leva à aprendizagem, como em qualquer outra habilidade prática, por imitação e método(exercício).

No raciocínio plausível trabalha-se com suposições e transitoriedade, envolvendo processos tais como: experiência, indução, generalização, especialização e analogia. Que a seguir serão apresentados:

Experiência

A experiência é o conhecimento de coisas pela observação, ela modifica o comportamento. Aprende-se ou deveria-se aprender com as experiências. O homem deveria fazer o maior uso possível da experiência.

Indução

A indução geralmente inicia com a observação. E leva intrinsecamente ao raciocínio que permite chegar a uma conclusão genérica do fato observado (ver figura 33).

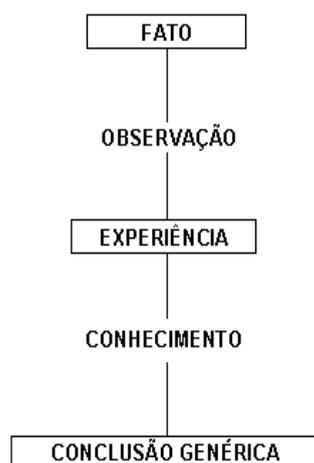
Por exemplo, um biólogo poderia observar a evolução da vida numa bactéria. Para que ele tenha chances de obter resultados interessantes de sua observação, deveria estar familiarizado com a evolução das bactérias. Similarmente, se um médico deseja obter algum resultado relevante ao observar a patologia do reumatismo, deveria estar familiarizado com ela. Porém, para distinguir a artrite reumatóide do lupus eritematoso, deveria conhecer os sintomas e as características particulares destas patologias. Isto é, o observador deve estar embasado em alguma teoria a respeito daquilo que observa.

Generalização

É passar desde a consideração de um conjunto de objetos a um conjunto maior que contenha o anterior (ver figura 34).

Por exemplo, o médico generaliza quando passa de sintomas característicos de diferentes patologias das articulações ao reumatismo. Frequentemente generalizamos quando passamos de sintomas característicos à uma patologia que contém os sintomas.

FIGURA 33 - PROCESSO INDUTIVO



Especialização

É passar desde a consideração de um conjunto de objetos a um conjunto menor que está contido no anterior (ver figura 35).

Por exemplo, o médico faz um processo de especialização quando passa desde a consideração de sintomas das doenças das articulações aos sintomas específicos que caracterizam cada tipo de patologia: lupus eritematoso, artrite reumatóide, gota, poliarterite, etc. Muito frequentemente nos especializamos ao passar de uma classificação geral de doenças a uma classe específica da patologia.

Analogia

Informalmente pode-se dizer que analogia é um tipo de similaridade. Esta similaridade se dá em um nível claramente definido e conceitual. Na analogia toma-se somente as características relevantes que identificam o objeto em consideração, sendo

estas características definidas de acordo com as intenções do observador. Casos similares possuem entre si alguns aspectos comuns. Quando se faz a redução dos aspectos comuns para conceitos claramente definidos, considera-se estes casos similares como análogos.

FIGURA 34 - PROCESSO DE GENERALIZAÇÃO



Dois sistemas são análogos, se forem similares em relações claramente definidas de suas respectivas partes. Se um fato novo observado é similar a outro já conhecido e sobre o qual já se tem uma opinião, então pode-se ter uma opinião sobre o fato novo baseado na experiência anterior, isto é, chega-se a um resultado por analogia (ver figura 36).

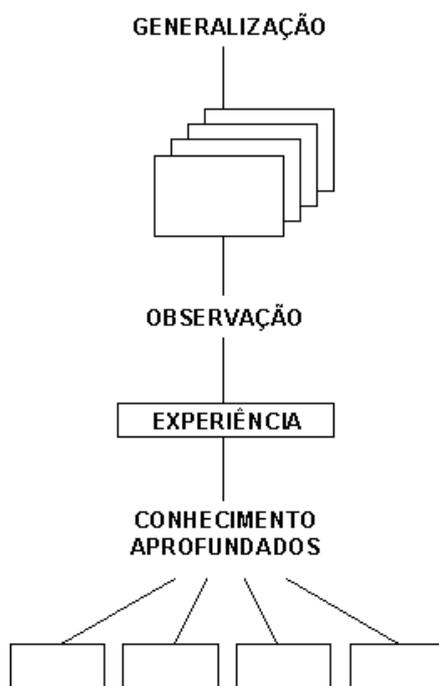
Por exemplo, um reumatologista iniciante encontra-se com o seguinte quadro clínico: febre, dor sem deformidades em uma ou mais articulações periféricas, ele lança um diagnóstico inicial: “trata-se de um caso de Lupus Eritematoso Sistêmico (LES)”, ele fundamenta essa conjectura por indução. Isto é, foi sugerida pela observação e indicado para este caso particular.

Estes sintomas observados no paciente não são suficientes para ele chegar ao diagnóstico; pois não tem muita experiência em LES para acreditar na sua conjectura inicial, mas vai se apoiar nas informações que possui de casos já vistos. Agora, tentar-se-á explicitar os passos feitos no raciocínio clínico: O médico começa observando os

sintomas e sinais da patologia, para ter experiência a respeito. A seguir, ele generaliza os sintomas e sinais chegando à conclusão de que se trata, neste exemplo, de uma patologia das articulações. Depois, ele vai especificar e identificar cada uma das doenças contidas de patologia das articulações. Finalmente, ao voltar para comparar com outros pacientes que tiveram sintomas similares, com alguma variação na intensidade ou duração, ele estará fazendo analogia.

Como o objetivo de resumir raciocínio plausível, apresentam-se os padrões da inferência plausível de George Polya [58]. Dentre aqueles são considerados de maior ênfase os seguintes:

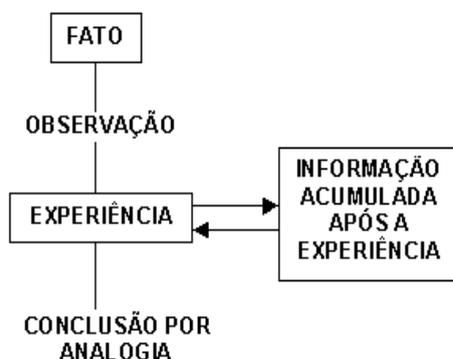
FIGURA 35 - PROCESSO DE ESPECIALIZAÇÃO



Padrões Indutivos: “A verificação de uma consequência torna a conjectura mais credível”. Uma conjectura pode ser escrita como a premissa numa regra contida no processo do raciocínio. Por exemplo, a conjectura “choveu ontem a noite” chega a ser mais credível quando verifica-se a consequência “o chão está molhado”. Sendo ambos considerados na seguinte regra: “Se choveu ontem a noite então o chão deve estar molhado”.

Verificação Sucessiva de Várias Conseqüências: “A verificação de uma nova conseqüência vale mais ou menos se a nova conseqüência difere mais ou menos da anterior, conseqüência verificada”. Por exemplo, ao se tentar verificar a conjectura “Todas as bananas tem casca amarela”, observa-se n bananas colombianas, todas elas de casca amarela. Nossa subsequente confiança nesta conjectura incrementa-se substancialmente se a $(n + 1)$ -ésima banana é uma banana brasileira mais amarela do que a outra banana da Colômbia. Tal não ocorreria se esta $(n + 1)$ -ésima banana fosse uma banana figo com sua casca escura.

FIGURA 36 - PROCESSO DE ANALOGIA



Verificação de Conseqüências Verossímeis: “A verificação de uma conseqüência vale mais ou menos, de acordo se a conseqüência é mais ou menos verossímil em si mesmo”. Por exemplo, tem-se duas expressões: “Se ontem a noite choveu, então o teto está pingando” e “Se ontem a noite choveu, então a grama está molhada”, pode-se perceber que o conseqüente da primeira expressão tem mais suporte do que o conseqüente na segunda expressão.

Inferência Desde Analogia: “A conjectura chega a ser mais credível quando uma conjectura análoga resulta ser verdadeira”. Por exemplo, a conjectura “O paciente apresenta queda de cabelo, febre e eritema facial, ele pode ter LES” chega a ser mais credível quando provamos a conjectura relacionada “Um dos quadros clínicos característicos do LES é: alopecia, febre e eritema facial”.

Polya questionou-se sobre se estes padrões poderiam ser considerados como regras do raciocínio plausível. Vejamos, algumas pessoas desconfiam das “regras” e

confiam em sua “intuição” ou “experiência” ou “intuição-e-experiência” [58]. Por exemplo:

Um cirurgião pode não ter regras na tomada de decisão, a decisão dele pode custar uma vida e algumas vezes, quando um paciente começa a sangrar ou a se sufocar, a decisão deve ser tomada em alguns segundos. Porém ele não usa regras. O tempo é muito curto para aplicar regras apropriadamente, e qualquer conjunto de padrões poderia enganá-lo.

Apresenta-se a mesma evidência a dois jurados e seguindo o mesmo procedimento eles podem julgar diferentemente: um pensa que a evidência é suficiente à prova do defendido e o outro pensa que não. Essas diferenças podem ter diferentes bases: um deles pode ser um ótimo observador de comportamento, ele observa as expressões do rosto, ele é impressionado pelos movimentos de olhos e mãos do acusado. O outro jurado pode não ser um bom observador, ele compreende melhor o meio social e as circunstâncias das pessoas envolvidas no caso. E assim eles chegam a resultados diferentes. Isto é, duas pessoas com a mesma evidência podem discordar honestamente.

Duas pessoas encontram-se interessadas na conjectura A (Uma delas é engenheiro, e A é uma conjectura relacionada à engenharia). Ambas sabem que A implica em B. Depois que elas acham que B é consequência de A, e B é verdadeiro. Elas concordam que A implica B, mas discordam sob o valor ou peso da evidência. Elas são honestas, mas a base de conhecimento de ambas é diferente, uma delas esta mais relacionada ao fato e a outra não. Observa-se que, apresentada a mesma evidência a duas pessoas e aplicando os mesmos padrões de inferência plausível, eles podem discordar honestamente.

Polya tentou olhar o raciocínio plausível no trabalho, no comportamento das pessoas em frente a problemas concretos. E com os pontos acima expostos, tentou aclarar a idéia na qual seus padrões estão “sujeitos”, e quão longe eles podem ser lembrados como “regras”.

Pode-se dizer que o raciocínio clínico para a tomada de decisão, é um raciocínio plausível, posto que se baseia em evidências que levam a conjecturas (o diagnóstico) que às vezes podem ser arriscadas, controversiais e transitórias, mas ainda assim não

criam dúvidas ao médico e não impedem a tomada de decisão. O raciocínio plausível encontra-se embutido no raciocínio clínico, o qual atravessa etapas desde a indução (médico iniciante) até a analogia (médico especialista).

3.5.6 - Raciocínio Não-Monotônico

Os sistemas tradicionais baseados na lógica de predicados são *monotônicos* no sentido de que o número de declarações conhecidas como sendo verdadeiras aumenta estritamente com o tempo. Novas declarações podem ser acrescentadas ao sistema e novos teoremas serem provados, mas nenhum desses eventos jamais fará com que uma declaração anteriormente conhecida ou provada torne-se não válida. Há diversas vantagens em trabalhar dentro de um sistema como este:

Quando uma nova declaração for acrescentada ao sistema, nenhuma conferência precisa ser feita para ver se há inconsistências entre a nova declaração e o conhecimento antigo;

Não é necessário lembrar, para cada declaração que tenha sido provada, a lista de outras declarações em que a prova repousa, pois não há perigo de essas declarações desaparecerem.

Infelizmente, tais sistemas monotônicos não são muito bons para lidar com três tipos de situações que muitas vezes surgem nos domínios de problemas reais: informação incompleta, situação em mudança e geração de suposições no processo de resolver problemas complexos.

3.5.6.1 - Introdução ao Raciocínio Não-Monotônico

Raramente um sistema tem a sua disposição toda informação que lhe seria útil. Mas, muitas vezes, quando tal informação estiver faltando, algumas estimativas sensatas podem ser feitas, desde que não ocorra nenhuma prova contraditória. A construção dessas estimativas é conhecida como raciocínio por omissão.

Por exemplo, suponhamos que você está a caminho da casa de um conhecido para jantar e passa por um vendedor de flores de beira de estrada. Seu anfitrião gostaria de receber flores?. Você provavelmente não possui qualquer informação específica para

responder a essa pergunta. Mas você poderá sair-se muito bem se utilizar uma regra geral que diz que como a maioria das pessoas gosta de flores, supõe-se que determinada pessoa goste, a menos que se possua provas em contrário (como uma alergia conhecida). Este tipo de raciocínio por omissão é não-monotônico (isto é, o acréscimo de mais uma informação poderá forçar a eliminação de outra) porque as declarações que forem derivadas desta forma dependem da falta de crença em outras declarações. Isto significa que se uma dessas declarações anteriormente faltantes for acrescentada ao sistema, a declaração gerada por omissão terá de ser eliminada. Assim, em nosso exemplo, se você chegar à porta com as flores na mão e seu anfitrião imediatamente começar a espirrar, você deveria eliminar sua crença anterior de que esta pessoa em particular gosta de flores. É claro que você também deverá eliminar quaisquer outras crenças que estiverem baseadas na que acabou de ser descartada.

O exemplo anterior ilustra um tipo comum de raciocínio por omissão, que podemos denominar opção mais provável. Sabemos que uma entre um conjunto de coisas deverá ser verdadeira e, na ausência de informação completa, optamos pela mais provável. A maioria das pessoas gosta de flores. A maioria dos cães possui rabo. A cor de cabelo mais comum para os suecos é o louro. Outro tipo importante de raciocínio por omissão é a circunscrição, McCarthy [50], onde supomos que apenas objetos que satisfaçam alguma propriedade P são os que podem ser demonstrados como satisfazendo-a. Por exemplo, suponhamos que estejamos tentando resolver um problema de remar um barco através de um rio. Há muitas coisas que poderiam impedir a utilização bem-sucedida do barco, incluindo falta de remos, furo no barco, lama excessiva no rio, e assim por diante. É importante que o programa de solução do problema não precise provar explicitamente que cada uma dessas condições não é verdadeira. Provavelmente a declaração do problema não fez citação a remos. O que o programa pode fazer é supor que apenas aquelas coisas que se possa provar explicitamente sejam verdadeiras (espera-se que nenhuma) e que o resto não é. Depois ele pode continuar e supor que pode utilizar o barco.

Uma descrição computacional precisa do raciocínio por omissão deverá relacionar a falta de alguma informação X a uma conclusão Y . O que queremos dizer é algo como:

Raciocínio por Omissão: Definição 1

Se X não for conhecido, então concluir Y.

Mas em todos os sistemas, excetuados os mais simples, apenas uma fração muito pequena das coisas que podem ser consideradas como conhecidas estão armazenadas explicitamente na base de dados. As outras podem, com graus variados de esforço, serem provadas a partir das coisas que são explícitas.

Raciocínio por Omissão: Definição 2

Se X não puder ser provado, então concluir Y.

Mas, supondo que ainda estamos trabalhando na lógica de predicados, como saberemos que X não pode ser provado?. O sistema não é capaz de decidir. Assim, não podemos garantir, para qualquer X arbitrário, que ele possa ser provado ou não.

Raciocínio por Omissão: Definição 3

Se X não puder ser provado em alguma quantidade de tempo alocada, então concluir Y.

Mas agora observe que a definição do processo de raciocínio pelo qual Y foi derivado depende de algo fora do escopo da lógica; depende da quantidade de computação que poderá ser feita no tempo alocado, e da eficiência dessa computação na busca da prova desejada. De modo que agora é essencialmente impossível fazer declarações formais a respeito do comportamento de nosso sistema. Ademais, perdemos a capacidade que tínhamos, na lógica de predicados, de conferir a correção de uma prova proposta, mesmo no caso de não haver garantia de encontrar tal prova, caso ela existisse. Podemos agora receber uma prova com um passo que diga que Y foi concluído por causa de falta de capacidade de provar X. Mas como não é possível decidir se X pode ser provado, a validade da prova maior em que ela está embutida também não é passível de decisão. Assim, a necessidade de raciocínio por omissão, que surge da falta de informação completa, nos força a utilizar sistemas cujo comportamento não pode ser caracterizado formalmente, com facilidade.

Mesmo se formos afortunados a ponto de algum dia possuir informações completas a respeito de uma situação, não é provável que continuemos nessa posição durante muito tempo, pois o mundo muda rapidamente ao nosso redor. Isto significa que declarações que eram completamente precisas em uma ocasião poderão não ser em outra. Infelizmente, esta solução não é perfeita, pois requer uma declaração distinta a respeito de cada estado onde cada predicado for verdadeiro. Assim, muito esforço será

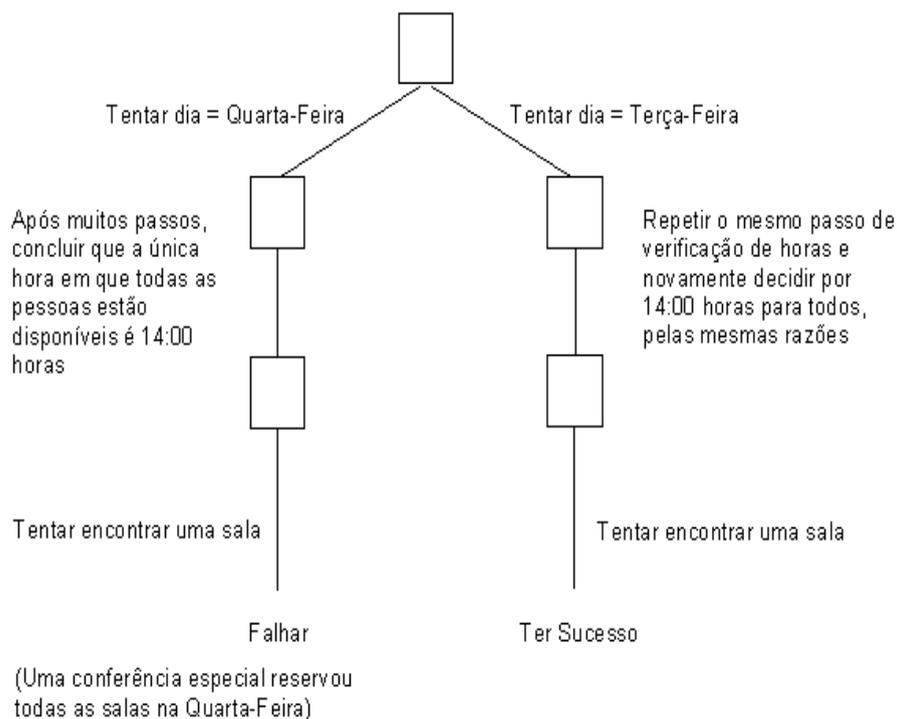
gasto ao se asseverar repetidas vezes o mesmo fato em lenta mutação. Outra maneira de resolver o problema de um mundo em mudanças é simplesmente eliminar declarações quando elas não mais descreverem o mundo com precisão, substituindo-as por outras mais precisas. Isto novamente nos leva a um sistema não-monotônico em que as declarações podem ser eliminadas bem como acrescentadas à base de conhecimento. E, novamente, cada vez que uma declaração for eliminada, outras declarações, cujas provas dependeram dela, também precisarão ser eliminadas.

Mesmo se o conhecimento disponível em um sistema não estiver sujeito a quaisquer dos problemas que acabamos de discutir, um bom sistema de solução de problemas poderá gerar, durante o processo de resolução, algum conhecimento que se comporte não-monotonicamente. Suponhamos que se queira construir um programa que gere uma solução para um problema relativamente simples, tal como encontrar um horário em que três pessoas ocupadas possam participar de uma reunião. Uma maneira de resolver esse problema é fazer primeiro uma suposição de que a reunião será realizada em algum dia determinado, digamos quarta-feira. Acrescenta-se à base de dados uma afirmativa neste sentido, devidamente rotulada como uma suposição, e depois tenta-se encontrar um horário, conferindo pelo caminho quaisquer inconsistências nas programações horárias das pessoas. Se surgir um conflito, a declaração representando a suposição deverá ser descartada e substituída por outra, que esperamos não seja contraditória. Mas, é claro, quaisquer declarações que forem geradas pelo caminho que dependam da suposição ora descartada também deverão ser descartadas. Assim, novamente temos um sistema não-monotônico.

Naturalmente, este tipo de situação pode ser tratado por uma busca de árvore direta com volta atrás. Todas as suposições bem como as inferências tiradas delas são registradas no nó de busca que as criou. Quando for gerada uma inconsistência, simplesmente volte atrás para o nó seguinte, do qual ainda existam caminhos inexplorados. As suposições e suas inferências desaparecerão automaticamente. A desvantagem desta abordagem está ilustrada na figura 37, que apresenta parte da árvore de busca de um programa que está tentando marcar uma reunião. Para fazê-lo, o programa deverá resolver um problema de satisfação de restrição para encontrar dia e

hora em que nenhum dos participantes estará ocupado, bem como uma sala suficientemente adequada e disponível.

FIGURA 37 - VOLTA ATRÁS DIRECIONADA À INDEPENDÊNCIA



Para resolver o problema, o sistema deverá tentar satisfazer uma restrição de cada vez. Inicialmente, não há razão para escolher uma alternativa sobre a outra, de modo que ele decide programar a reunião para quarta-feira. Isso cria uma nova restrição que deverá ser satisfeita pelo resto da solução. A suposição de que a reunião será realizada na quarta-feira é armazenada no nó que ela gerou. A seguir o programa tenta escolher uma hora em que todos os participantes estão disponíveis. Todos eles têm reuniões diárias regularmente marcadas todas as horas, exceto às 14:00h. Assim, 14:00h é escolhida como a hora para a reunião. Mas não teria feito diferença alguma o dia que fosse escolhido. Então o programa descobre que na quarta-feira não há sala disponível. Assim, ele volta atrás além da suposição da quarta-feira e tenta outro dia, terça-feira. Agora ele deverá duplicar a cadeia de raciocínios que o levou a optar por 14:00h. Isto ocorreu muito embora esse raciocínio não dependesse de modo algum da suposição de que o dia seria quarta-feira.

Ao retirar as declarações, baseados na ordem em que elas foram geradas pelo processo de busca, em vez da responsabilidade pela inconsistência, poderemos perder muito tempo e esforço. Assim, seria bom se fôssemos capazes de inserir diretamente suposições na base de dados e de retirá-las como também quaisquer inferências nela baseadas, conforme necessário. Esse processo foi denominado de “Volta Atrás Direcionada à Dependência”, Stallman [69].

Os sistemas de raciocínio não-monotônicos poderão ser necessários por qualquer uma das seguintes razões:

A presença de informações incompletas requer um raciocínio por omissão;

Um mundo em mudanças deverá ser descrito por uma base de dados em mutação;

A geração de uma solução completa para um problema poderá exigir suposições temporárias a respeito de soluções parciais.

É mais difícil lidar com sistemas não-monotônicos do que com sistemas monotônicos porque, muitas vezes, quando uma declaração for eliminada da base de conhecimento, é necessário voltar atrás sobre outras declarações cujas provas dependem da declaração eliminada, e eliminá-las ou encontrar novas provas que sejam válidas em relação à atual base de conhecimento. A eliminação de uma única declaração poderá determinar um efeito significativo em toda base de conhecimento, já que todas as provas que dela dependem deverão ser eliminadas; a seguir, todas as provas que dependem das declarações cujas provas acabaram de ser descartadas (e para as quais nenhuma outra prova pode ser encontrada) também deverão ser dispensadas, e assim por diante. Ao projetar sistemas não-monotônicos, é importante assegurar que o sistema não gaste todo o tempo propagando mudanças.

Para ser capaz de propagar mudanças na base de dados e conferir provas quanto a sua validade atual, é importante armazenar, junto com cada teorema, sua prova, ou pelo menos uma lista das outras declarações das quais a prova depende. Isto não é necessário em sistemas monotônicos, pois, uma vez encontrada uma prova, ela nunca precisará ser reexaminada. Assim, os sistemas não-monotônicos poderão exigir mais espaço de

armazenamento, bem como mais tempo de processamento, do que os sistemas monotônicos.

3.5.7 - Raciocínio Estatístico e Probabilístico

Há três tipos de situações em que é tentador utilizar o raciocínio probabilístico:

Mundo relevante é realmente aleatório, por exemplo, o movimento dos elétrons no átomo ou a distribuição das pessoas que ficarão doentes durante uma epidemia.

Mundo relevante não é aleatório, dadas as informações suficientes, mas nosso programa nem sempre terá acesso a todos esses dados. Por exemplo, a probabilidade de sucesso de um remédio para combater uma doença num determinado paciente.

Mundo parece ser aleatório porque não o descrevemos ao nível certo, por exemplo, a tarefa de reconhecimento de padrão.

Entretanto, no terceiro caso, deveríamos primeiro ver se não podemos descobrir um modelo melhor do mundo em que o raciocínio probabilístico não fosse mais necessário. Por exemplo:

Considere o problema de decidir qual carta jogar, no jogo de bridge. Estamos lidando com um mundo genuinamente aleatório. Nenhuma quantidade de conhecimento adicional (excluindo o roubo) nos permitiria descrever a situação completamente. Assim, teremos de utilizar o raciocínio probabilístico.

Consideremos, a seguir, o problema de diagnosticar as doenças das pessoas a partir de anotações clínicas. Provavelmente, há alguma aleatoriedade no mundo relevante e, também, em nossa descrição do mundo, pois a ciência médica não domina completamente o funcionamento do corpo humano. E, além disso, devemos projetar um programa que possa funcionar mesmo não tendo acesso a todos os dados que a ciência médica possa fornecer, já que alguns testes clínicos são caros e perigosos. Para se trabalhar com dados particularmente tão incompletos, teremos de utilizar o raciocínio probabilístico.

IV - SISTEMA ESPECIALISTA PARA PRIMEIROS SOCORROS PARA CÃES (SEPRISCA)

4.1 - INTRODUÇÃO

Nos últimos anos tem crescido bastante o número de trabalhos e pesquisas de ferramentas computacionais em Inteligência Artificial que procuram capturar e simular o comportamento de especialistas humanos. A construção de um software para o desenvolvimento de sistemas especialistas - *shell* - não é trivial, tendo em vista que este software deve implementar o tratamento de problemas complexos do mundo real que necessitam da interpretação de um especialista. A necessidade de utilização de sistemas especialistas deve-se a diversos fatores tecnológicos e econômico-sociais, dentre os quais temos: a dificuldade de acesso a especialistas humanos em determinadas regiões, o armazenamento e formalização do conhecimento de vários especialistas humanos, ferramenta de apoio à tomada de decisões por parte do especialista, treinamento de profissionais e imparcialidade na tomada de decisões.

A arquitetura mais comum de sistemas especialistas é a que envolve regras de produção (*production rules*). Essas regras são simplesmente um conjunto de condições no estilo SE... ENTÃO..., com a possibilidade de inclusão de conectivos lógicos relacionando os atributos no escopo do conhecimento e o uso de probabilidades, como vemos no exemplo a seguir:

SE olho fora de órbita = Sim
OU superfície do globo ocular = Sim
OU sangue ou algum corpo estranho = Sim
ENTÃO

Problema = Afecções dos olhos: Primeiros socorros. Vá ao veterinário já CNF 100%

Um sistema de bom tamanho tem em torno de centena de regras (considerando aqueles que utilizam regras). Visando uma maior viabilidade econômica na implementação de um sistema especialista, e considerando-se que diversos sistemas

compartilham uma *máquina de inferência* e outras características comuns de ambiente, foram criadas ferramentas, *shells*, aptas a realizar muito do trabalho necessário para transpor um sistema especialista para um computador. Essas ferramentas permitem que o criador do sistema preocupe-se somente com a representação do conhecimento do especialistas, deixando para a *shell* a tarefa de interpretar o conhecimento representado e executá-lo em uma máquina, além de permitir depurações e explicações de como o computador chegou àquela(s) conclusão(ões). A principal função de uma *shell* é simplificar ao máximo o trabalho de implementação de um sistema especialista e permitir seu uso por qualquer pessoa sem conhecimentos de informática.

O Expert SINTA é uma ferramenta computacional que utiliza técnicas de Inteligência Artificial para geração automática de sistemas especialistas. Esta ferramenta utiliza um modelo de representação do conhecimento baseado em regras de produção e probabilidades, tendo como objetivo principal simplificar o trabalho de implementação de sistemas especialistas através do uso de uma máquina de inferência compartilhada, da construção automática de telas e menus, do tratamento probabilístico das regras de produção e da utilização de explicações sensíveis ao contexto da base de conhecimento modelada.

Implementado na linguagem de programação orientada a objetos Borland Delphi, possibilitando um suporte visual de fácil operação. O usuário responde a uma seqüência de menus, e o sistema encarregar-se-á de fornecer respostas que se encaixem no quadro apontado pelo usuário.

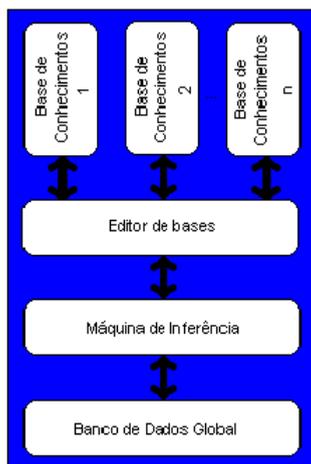
O Expert SINTA permite o desenvolvimento modular de bases de conhecimento através de uma interface de fácil manipulação e de utilitários criados para depuração.

Uma base de conhecimento no Expert SINTA envolve os seguintes conjuntos de atributos que devem ser indicados pelo projetista da base: variáveis, regras, perguntas, objetivos e informações adicionais. Quando esses elementos estiverem definidos, já se torna possível utilizar o sistema especialista.

4.2 - ARQUITETURA DE UM SISTEMA ESPECIALISTA NO EXPERT SINTA

Os sistemas especialistas (SE) gerados no Expert SINTA seguem a arquitetura abaixo:

FIGURA 38 - ARQUITETURA SIMPLIFICADA DO EXPERT SINTA



Onde:

Base de conhecimentos: representa a informação (fatos e regras) que um especialista utiliza, representada computacionalmente.

Editor de bases: é o meio pelo qual a shell permite a implementação das bases desejadas.

Máquina de inferência: é a parte do SE responsável pelas deduções sobre a base de conhecimentos.

Banco de dados global: são as evidências apontadas pelo usuário do sistema especialista durante uma consulta.

4.3 - GERENCIANDO BASES

Como padrão, O Expert SINTA grava as bases de conhecimento geradas em arquivos *.BCM. No menu Arquivo tem-se as seguintes opções:

Abrir base: abre bases pré-existentes em seus discos.

Nova base: cria uma base totalmente nova.

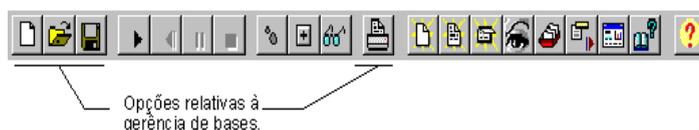
Salvar base: salva a base de conhecimento que está em uso.

Salvar como: cria uma base de conhecimento igual aquela que está em uso, porém com outro nome.

Imprimir: impressão da base de conhecimentos.

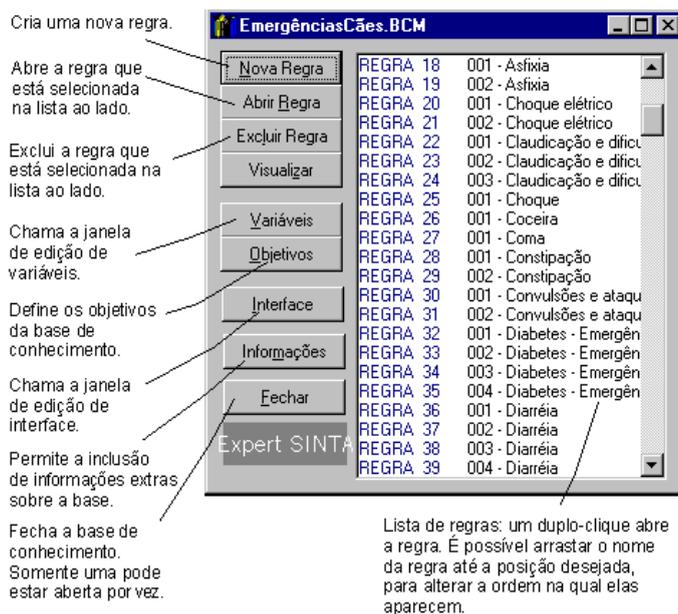
Muitas das opções dos menus encontram-se também na barra de ferramentas, inicialmente presente no topo da área de trabalho do Expert SINTA. É possível mudar a posição da barra de ferramentas clicando sobre ela e arrastando o mouse.

FIGURA 39 - BARRA DE FERRAMENTAS



4.3.1 - A Janela “Knowledge-in-a-box” (KIB)

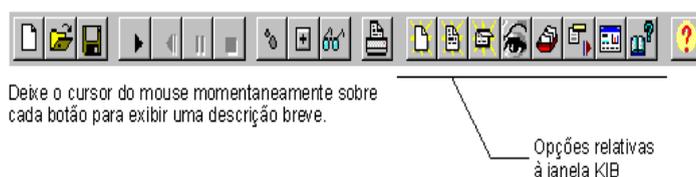
FIGURA 40 - JANELA “KNOWLEDGE-IN-A-BOX” (KIB)



As opções de edição de base da janela apresentada, e outras, encontram-se repetidas na barra de ferramentas, inicialmente presente no topo da área de trabalho do Expert SINTA.

Deixe o cursor do mouse momentaneamente sobre cada botão para exibir uma descrição breve.

FIGURA 41 - BARRA DE FERRAMENTAS

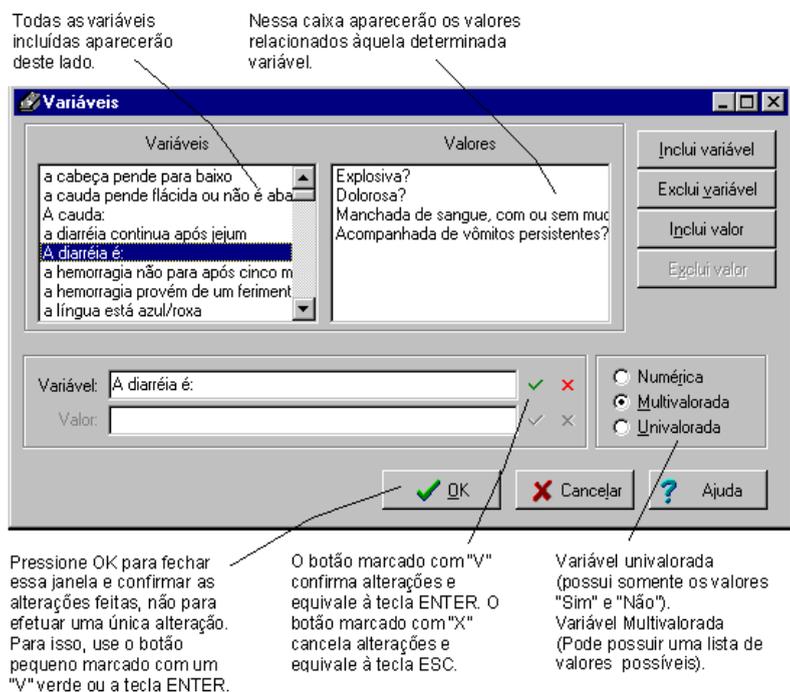


4.3.1.1 - Criando Variáveis

Antes de podermos criar regras, é necessário que todas as variáveis utilizadas, bem como seus respectivos valores, sejam criados. Através desse mecanismo, a base fica organizada, fácil de manter e as regras podem ser criadas visualmente.

Na janela da base de conhecimentos, clicando no botão Variáveis, aparecerá a seguinte janela:

FIGURA 42 - CRIANDO VARIÁVEIS



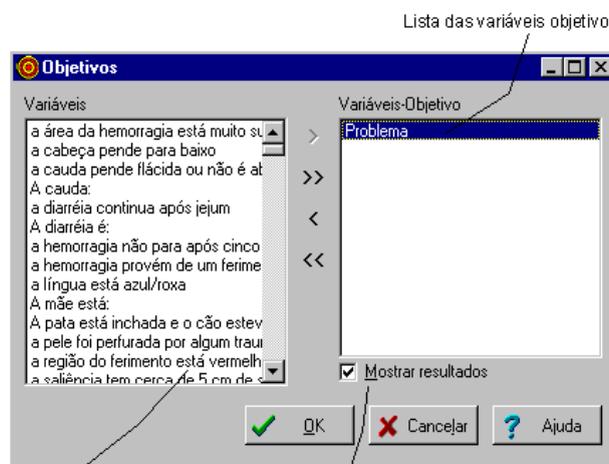
4.3.1.2 - Definindo o(s) Objetivo(s)

O objetivo de uma consulta a um especialista é encontrar a resposta para um determinado problema. Assim também o é um sistema especialista. A diferença é que, aqui, os “problemas” são representados por variáveis. Antes de podermos executar o sistema pela primeira vez, é preciso que definamos quais são as variáveis (chamadas *variáveis objetivo*) que irão controlar o modo como a máquina de inferência se comporta.

Na janela de base de conhecimentos, clicando no botão Objetivos, aparecerá uma janela com duas listas: a lista das variáveis comuns e a das variáveis objetivo.

Para fazer de uma variável comum uma variável objetivo e vice-versa, selecione a variável desejada e clique no botão correspondente, aquele que aponta para a lista de destino. A lista com uma seta dupla (seja para a esquerda ou para direita) move todos os itens de uma lista para outra.

FIGURA 43 - DEFININDO VARIÁVEIS-OBJETIVO



Lista das variáveis que podem ser utilizadas como objetivos do sistema especialista.

Deixe marcada esta caixa para que o Expert SINTA mostre em destaque os resultados atingidos do objetivo marcado acima.

4.3.1.3 - Trabalhando com Regras

O Expert SINTA utiliza regras de produção para modelar o conhecimento humano, o que o torna ideal para problemas de seleção, no qual uma determinada solução deve ser atingida a partir de um conjunto de seleções.

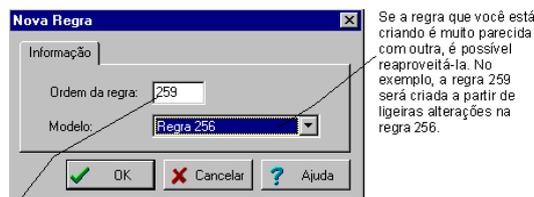
Na janela da base de conhecimentos tem-se as seguintes opções:

Nova regra: cria uma nova regra.

Abrir regra: abre uma regra.

Excluir regra: exclui uma regra.

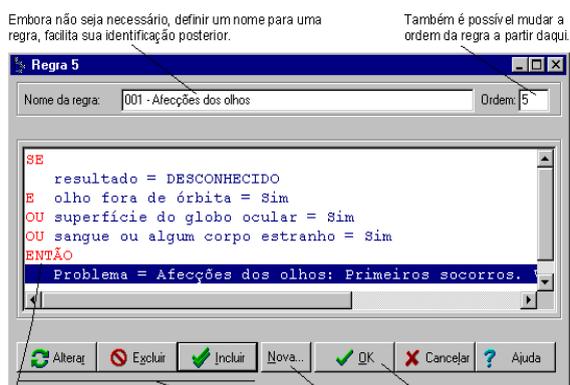
FIGURA 44 - CAIXA DE DIÁLOGO “NOVA REGRA”



Se a regra que você está criando é muito parecida com outra, é possível reaproveitá-la. No exemplo, a regra 259 será criada a partir de ligeiras alterações na regra 256.

Indique a posição dessa regra em relação às demais (pois a ordem das regras influencia o descobrimento de soluções).

FIGURA 45 - CRIANDO REGRAS



Embora não seja necessário, definir um nome para uma regra, facilita sua identificação posterior.

Também é possível mudar a ordem da regra a partir daqui.

É aqui que aparecem as sentenças da sua regra. A sentença selecionada será aquela que sofrerá alterações.

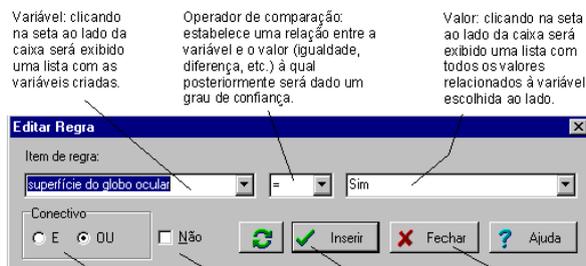
Cada um desses botões efetua uma operação diferente com a sentença selecionada.

Cria a próxima regra. A regra aberta no momento será fechada.

SOMENTE pressione esse botão QUANDO FINALIZAR a regra.

Para inserir uma premissa, deve-se marcar o local na caixa de listagem na qual se deseja inserir a sentença e clicar o botão Inserir. Uma outra janela aparecerá com várias listas.

FIGURA 46 - CRIANDO PREMISA(S) DA REGRA



Variável: clicando na seta ao lado da caixa será exibido uma lista com as variáveis criadas.

Operador de comparação: estabelece uma relação entre a variável e o valor (igualdade, diferença, etc.) à qual posteriormente será dado um grau de confiança.

Valor: clicando na seta ao lado da caixa será exibido uma lista com todos os valores relacionados à variável escolhida ao lado.

Conectivo: escolha o conectivo que irá ligar a sentença de regra às outras sentenças na mesma regra.

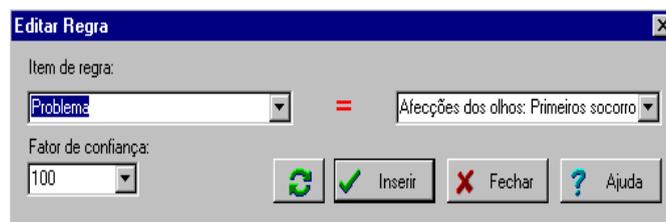
Negação: deixar marcada essa caixa para avaliar essa sentença por sua negativa. Equivale à inverter os operadores (transformar "=" em "<=>", "<=" em ">", etc.)

Inserir uma nova premissa ou conclusão. Você poderá inserir várias sem necessidade de fechar esta janela, contanto que as premissas/conclusões inseridas sejam consecutivas.

Deve-se "Fechar" quando terminar de inserir os itens desejados.

Para inserir uma conclusão, proceda de maneira análoga, escolhendo um item a partir do conectivo ENTÃO na sua regra. Em conclusões, somente o operador de atribuição (“=”) pode ser utilizado. A última lista presente indica o grau de confiança daquela atribuição. Deixar a lista vazia indica uma confiança de 100% (cem por cento).

FIGURA 47 - CRIANDO CONCLUSÃO DA REGRA



Para editar uma sentença, deve-se selecioná-la e clicar no botão Alterar (ou dê um duplo clique sobre ela). Para eliminar uma sentença, deve-se selecioná-la e clicar no botão Excluir.

FIGURA 48 - EDITANDO PREMISA DA REGRA

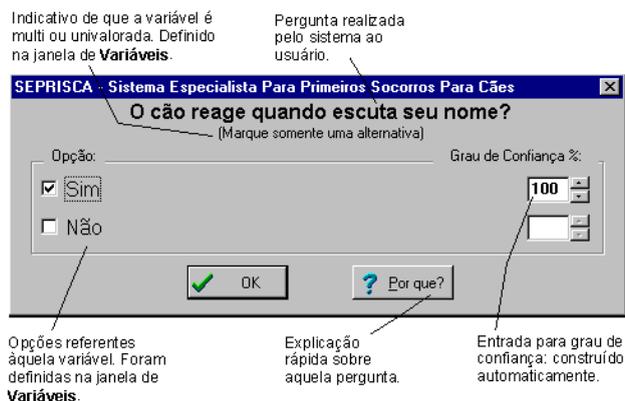


Observação: também é possível visualizar regras em conjunto, clicando no botão Visualizar da janela KIB.

4.3.1.4 - Definindo Interface com o Usuário

Um sistema especialista implementado com o Expert SINTA comunica-se com o usuário final através de menus de múltipla escolha (ou escolha simples, se a variável em questão for univalorada). Estes menus são construídos automaticamente pela *shell*, mas alguns detalhes devem ser fornecidos pelo criador da base. Os elementos da interface com o usuário são os seguintes:

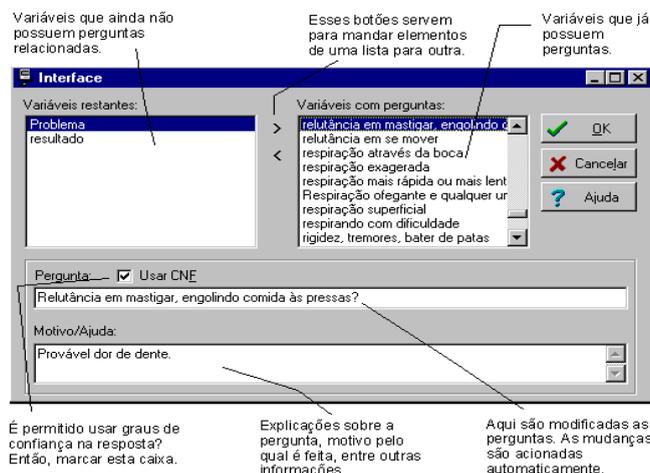
FIGURA 49 - INTERFACE COM O USUÁRIO



A pergunta realizada pela máquina de inferência deve ser personalizada para que seja inteligível. Quando o Expert SINTA necessita perguntar o valor de determinada variável, ele cria uma sentença genérica do tipo “Qual o valor de *x*?”, onde *x* é o nome da variável. Deve-se criar uma sentença mais clara através do editor de interface.

Para isso, clicando no botão Interface da janela KIB. O Expert SINTA permite modificações em apenas dois itens: perguntas e explicações rápidas.

FIGURA 50 - DEFININDO A INTERFACE COM O USUÁRIO



Para visualizar as perguntas das variáveis, deve-se clicar em alguma delas, na lista à direita. Para modificar a pergunta ou o motivo, simplesmente reescreva o conteúdo das caixas de texto. As mudanças são automáticas. Para eliminar uma pergunta, simplesmente mova o item selecionado para a lista da esquerda, pressionando o botão correspondente;

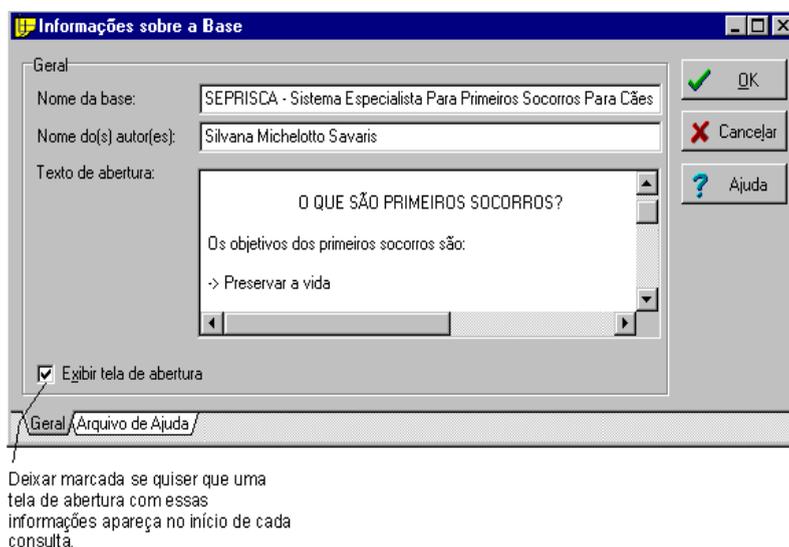
Para criar uma nova pergunta, selecione a variável desejada, na lista à esquerda, e mova-a para a lista à direita, pressionando o botão correspondente, e digite o conteúdo das caixa de texto. As mudanças são automáticas.

4.3.1.5 - Informações Adicionais sobre a Base

A janela “Informações sobre a Base”, exibida a partir do botão Informações da janela KIB, permite a definição de informações esclarecedoras sobre a base, indicação dos autores e definição de contextos de ajuda que podem ser vitais para o aproveitamento do sistema especialista. A janela se divide em duas partes:

Geral: Aqui, informações iniciais sobre o sistema são incluídas, tais como o nome da base, os autores e um texto de abertura. Todos os dados aparecerão em uma tela de abertura quando a base é colocada em funcionamento. Para permitir a exibição desses dados, deve-se deixar marcada a caixa de verificação “Exibir tela de abertura”, na parte inferior da janela.

FIGURA 51 - CAIXA DE DIÁLOGO “INFORMAÇÕES SOBRE A BASE”

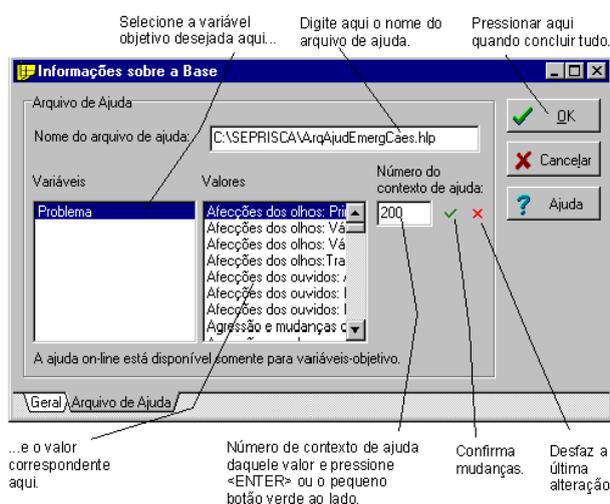


Arquivo de Ajuda: um bom sistema especialista não deve saber somente encontrar respostas, mas também fazer o usuário utilizar a resposta com fins práticos. Para isso, o Expert SINTA possibilita a inclusão de tópicos de ajuda para os valores possíveis de determinados atributos, associando a um arquivo no formato Ajuda do Windows a sua base.

A idéia desses textos explicativos não é somente ativar uma ajuda on-line quando o usuário estiver utilizando o sistema, mas ir diretamente ao tópico explicativo de uma resposta, quando o Expert SINTA atinge um determinado objetivo. Para isso, o desenvolvedor deve, na janela Informações sobre a Base, escolher os valores das variáveis-objetivo que irão possuir um número de contexto de ajuda. Quando o usuário final ativar a ajuda on-line, ela aparecerá diretamente no tópico referente à solução específica. O desenvolvedor também deve incluir o nome do arquivo de ajuda na caixa de texto “Nome do arquivo de ajuda”, na mesma janela.

Para garantir melhor aproveitamento da sua base, textos didáticos podem ser relacionados a cada solução encontrada. Para tanto, deve-se criar um arquivo de ajuda no formato “Ajuda do Windows” e definir os contextos de ajuda relativos a cada valor (se desejado) de cada variável objetivo.

FIGURA 52 - DEFININDO TÓPICOS DE AJUDA PARA VALORES POSSÍVEIS DAS VARIÁVEIS-OBJETIVO



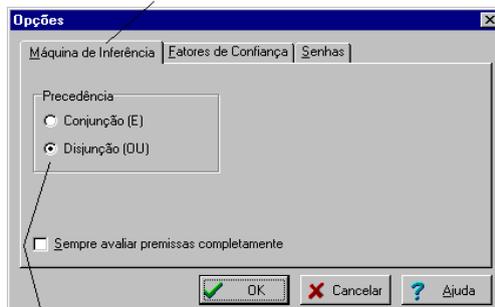
4.3.1.6 - Opções

Para permitir maior flexibilidade na construção de um sistema especialista, o Expert SINTA oferece diversas opções de configuração. Para acessá-las, deve-se utilizar o menu Exibir|Opções. Uma caixa de diálogo com três partes aparecerá.

Máquina de Inferência: permite pequenas modificações no comportamento da máquina de inferência, como o modo pelo qual as premissas são avaliadas.

FIGURA 53 - DEFININDO O MODO DE AVALIAÇÃO DAS PREMISSAS PELA MÁQUINA DE INFERÊNCIA

Dependendo do modo como a precedência for organizada ou como os fatores de confiança forem definidos, a máquina de inferência pode saber que uma regra não vai atingir o grau mínimo de confiança para ser aceita antes mesmo de avaliá-la por completo. Se você desejar que as regras sejam sempre avaliadas completamente, mesmo quando já se sabe se serão aceitas ou não, deve-se marcar esta caixa.



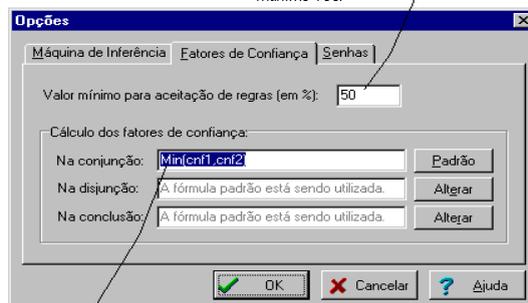
Defina aqui qual a precedência dos operadores lógicos E e OU. Lembre-se da diferença na expressão: **A e B** ou **C**, se mantivermos a precedência no "e", ela pode ser interpretada como **(A e B)** ou **C**. Se a precedência for do "ou", **A e (B ou C)**.

Fatores de Confiança: após avaliar todas as premissas de uma regra, o Expert SINTA pode aceitá-la ou rejeitá-la. Se, ao calcular o fator de confiança final ele for maior que o mínimo estipulado, a regra será aceita. É nessa janela que se define o valor mínimo. Também é aqui que se pode personalizar as funções de cálculo de fatores de confiança.

Maiores detalhes sobre Tratamento de Incerteza em Inteligência Artificial podem ser encontrados em: [33], [36], [38], [67], [68], [66], [77], [55], [45], [22], [3], [1], [44], [72], [28], [63], [71], [23], [34], [76].

FIGURA 54 - DEFININDO FATORES DE CONFIANÇA

Aqui é definido o valor de confiança mínimo para que uma regra seja aceita. Colocar na escala percentual: mínimo 0 e máximo 100.



O Expert SINTA utiliza algumas funções para calcular fatores de confiança em determinados eventos. Estes eventos podem ser uma conjunção, uma disjunção ou uma regra aceita. Pode-se usar as funções padrões do Expert SINTA ou criar novas. Cada função apresenta dois parâmetros implícitos, *cnf1* e *cnf2* (ou seja, o grau de confiança do primeiro termo da conjunção e o grau de confiança do segundo termo da conjunção, respectivamente). Para, por exemplo, mudar a função da conjunção para o mínimo entre os dois valores (deve-se pressionar o botão **Alterar** se a fórmula padrão estiver sendo utilizada) ao lado da função de conjunção e digite `Min(cnf1, cnf2)`. Para retornar à forma padrão, aperte o botão respectivo.

Senha: um sistema especialista não deixa de ser um programa de computador. Portanto, pode se fazer necessário algum tipo de proteção sobre este *software*.

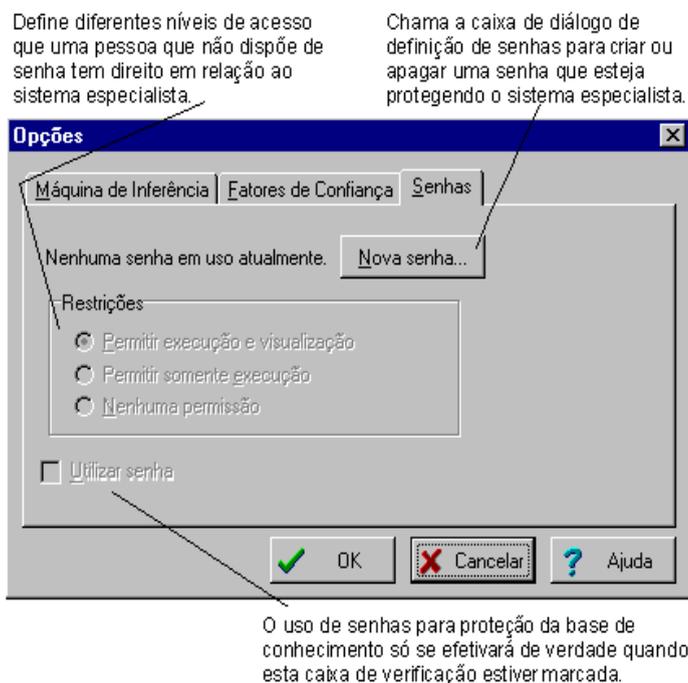
O Expert SINTA permite três níveis de proteção:

Permitir Execução e Visualização - O usuário do seu sistema especialista não tem permissão para modificar a base, mas pode executá-la, depurá-la e imprimi-la. É uma opção para evitar que estranhos modifiquem a base, fazendo com que perca sua consistência.

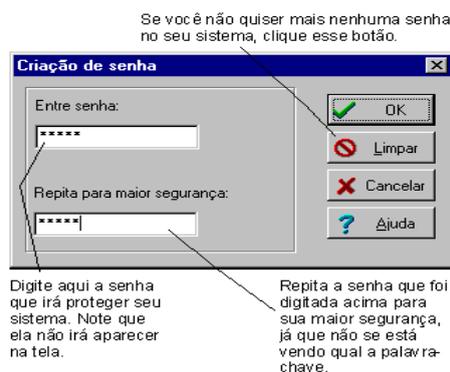
Permitir Somente Execução - Essa opção permite que o usuário sem senha apenas possa executar o sistema especialista, mas não pode modificar a base e menos ainda visualizá-la. O principal motivo que levaria um projetista a escolher essa opção de restrição é o perigo de pirataria do conteúdo do sistema especialista.

Nenhuma Permissão - Somente pessoas com senha podem utilizar esse sistema especialista.

FIGURA 55 - DEFININDO PERMISSÕES



Para criar uma senha, clique no botão Nova senha... (ele se chama Mudar senha... caso alguma senha já esteja sendo utilizada). Uma caixa de diálogo aparecerá:

FIGURA 56 - CRIANDO SENHA

4.4 - CONSULTANDO O SISTEMA ESPECIALISTA

O Expert SINTA mantém uma interface uniforme para a consulta de qualquer base de conhecimento criada em seu ambiente. Existem dois modos pelos quais podem ser efetuadas consultas:

A execução usual, na qual o usuário acompanha uma seqüência de menus de múltipla (ou única) escolha, nos quais deve-se indicar informações que resultarão nas conclusões atingidas pelo sistema especialista;

O modo de acompanhamento, pelo qual é possível examinar o conteúdo das regras que formam o sistema, bem como acompanhar a execução passo a passo e as instâncias (valores) que cada variável possui em determinado momento.

Para utilizar este modo, é preciso que o projetista da base não tenha protegido o acesso às regras por meio de uma senha.

Um sistema especialista procura atingir conclusões para determinados objetivos. Sempre que um desses objetivos é atingido, ou quando se esgotam todas as possibilidades, o Expert SINTA apresentará uma janela com os resultados e o acompanhamento de como se chegou àquela conclusão (também é necessário que o projetista não tenha restringido o acesso).

Alguns sistemas também devem trazer um sistema de ajuda on-line, pelo qual serão dadas maiores informações de como se deve melhor utilizar as respostas conseguidas, bem como explicações mais detalhadas sobre as funções daquele determinado sistema especialista.

4.4.1 - Conceitos Rápidos

Projetista do Conhecimento: é o encarregado de transportar o conhecimento humano para uma série de passos que um computador é capaz de entender.

Base de Conhecimento: é o conjunto de informações, representado no Expert SINTA na forma de regras SE-ENTÃO, as quais supostamente devem agir conforme um especialista humano. Seria, então, a “alma” do sistema especialista.

Variáveis: são os elementos do mundo real representados na base de conhecimento, como, por exemplo, um tipo de doença, a quilometragem de um carro, sintomas de um cão, a posse ou não de crédito em determinada agência bancária, etc.

Valores: são instâncias das variáveis, ou seja, uma variável pode possuir um ou mais valores (quem decide é o projetista do conhecimento). Também podem existir casos nos quais uma variável permanece DESCONHECIDA, ou seja, o sistema não conseguiu nenhum valor satisfatório para ela.

Depuração: acompanhamento da execução da consulta com o intuito de compreender como o sistema especialista chegou àquela(s) conclusão(ões). Também pode ser usada com o intuito de corrigir falhas na base de conhecimento.

4.4.2 - Guia Visual de Operações

Existem dois modos pelos quais se pode acompanhar uma consulta no Expert SINTA. Utilizando o menu da janela principal ou a barra de ferramentas. As operações de execução/depuração podem ser:

Iniciar a consulta;

Interromper a execução para saber que regra está utilizando naquele ponto;

Executar passo a passo;

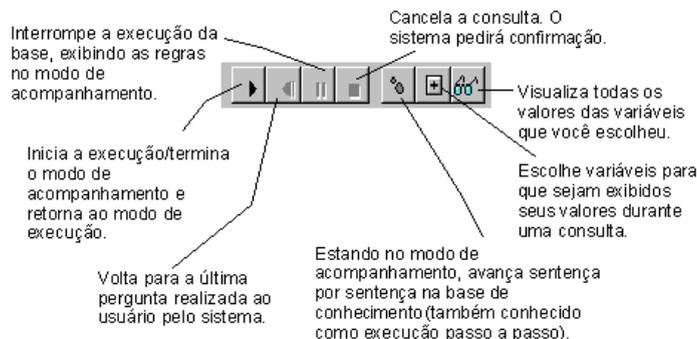
Adicionar elementos da base (variáveis) para acompanhar os valores que eles recebem durante a consulta;

Ver esses valores escolhidos;

Abandonar a consulta.

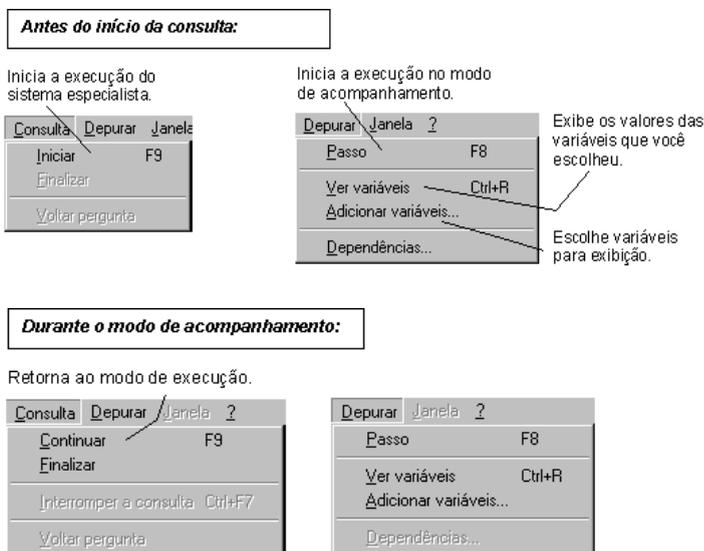
Os comandos equivalentes na barra de ferramentas são:

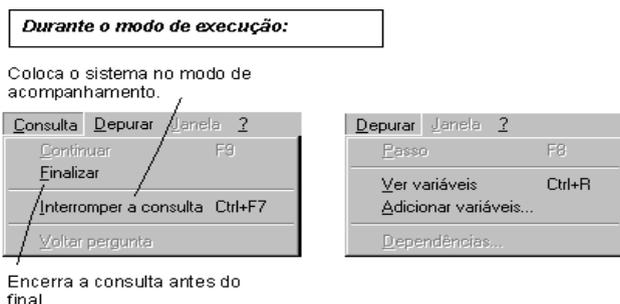
FIGURA 57 - A BARRA DE FERRAMENTAS PARA EXECUÇÃO DE CONSULTAS



A seguir, são especificados os comandos equivalentes através do menu. Aqueles que estiverem em cinza não estão disponíveis naquele momento. Alguns itens apresentam um nome representando uma tecla (ou uma combinação de teclas) que pode ser utilizada em substituição ao menu.

FIGURA 58 - OS ITENS DE MENU PARA A EXECUÇÃO DE CONSULTAS



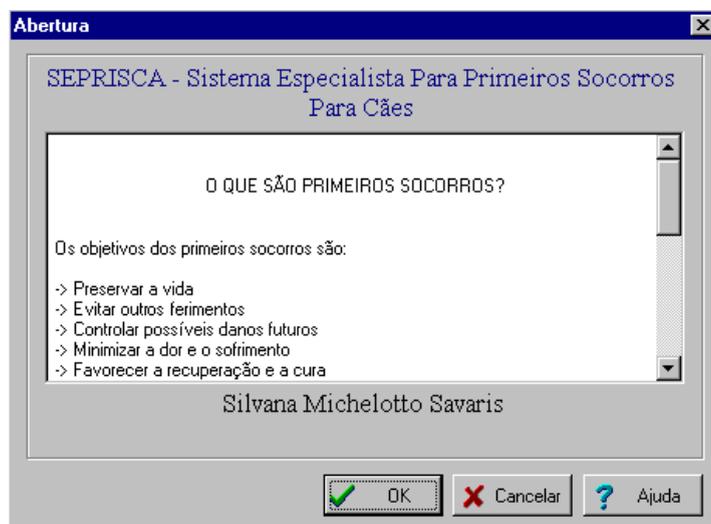


4.4.3 - A Consulta

Iniciando-se a consulta, seja no modo de execução ou acompanhamento, o sistema especialista poderá apresentar uma tela de abertura, com informações sobre seu funcionamento, os autores e casos aos quais ele se aplica, ou outra informação.

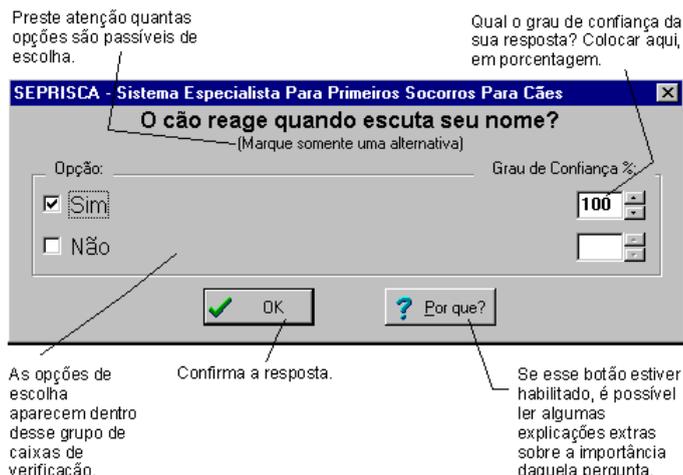
Para iniciar a consulta, deve-se pressionar o botão OK. O botão Cancelar abandona a execução do sistema especialista. O botão Ajuda chama a ajuda da base, se ela estiver disponível.

FIGURA 59 - UMA ABERTURA TÍPICA DE UM SISTEMA ESPECIALISTA



A consulta se desenvolve por meio de menus de múltipla (ou única) escolha.

FIGURA 60 - UMA PERGUNTA REALIZADA PELO SISTEMA AO USUÁRIO



Escolha a(s) opção(ões) desejada(s) marcando a(s) respectiva(s) caixa(s) de verificação, localizadas sempre à esquerda de cada alternativa. Observe, à direita, na figura 60, a possibilidade de entrar com o grau de confiança da sua resposta. Graus de confiança são utilizados quando não se possui certeza absoluta sobre um fato. Assim sendo, você pode expressar sua dúvida por meio de um número percentual, ou seja, a sua resposta terá validade no intervalo de zero a cem por cento. Cada resposta assinalada pode ter o seu próprio grau de confiança.

4.4.4 - As Janelas de Acompanhamento

Quando se está no modo de acompanhamento, uma nova janela sempre estará aberta, o depurador (figura 61).

FIGURA 61 - O DEPURADOR

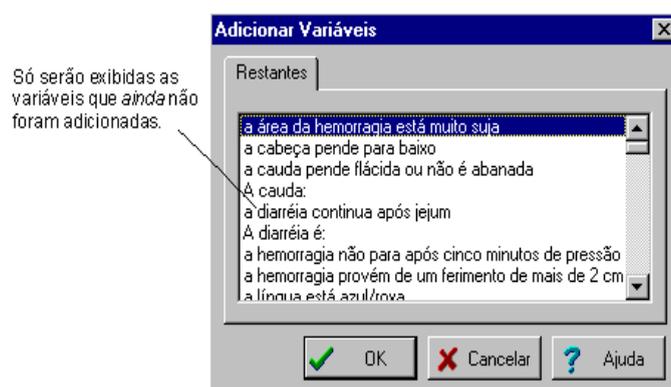


O depurador exibe, em sua caixa de listagem, todas as regras da base de conhecimento. A sentença em destaque é aquela que está sendo analisada no momento pela máquina de inferência. Observe que é possível rolar a lista para cima e para baixo, exibindo o resto da base, mas somente a linha representando o ponto de parada pode ser selecionada.

Utilizando o menu Depurar|Passo, ou o botão respectivo da barra de ferramentas, ou ainda a tecla F8, pode-se continuar a acompanhar, passo a passo, a execução do sistema. Se o sistema estiver realizando uma pergunta para você, primeiro responda antes de executar um passo, ou nada acontecerá.

Fechar o depurador equivale a voltar ao modo de execução.

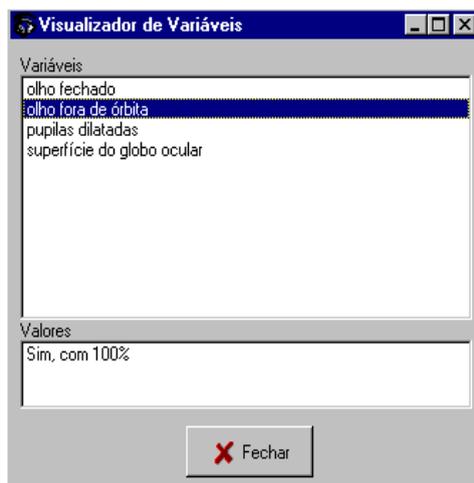
FIGURA 62 - CAIXA DE DIÁLOGO “ADICIONAR VARIÁVEIS”



Pode-se selecionar uma ou mais variáveis. Mantendo pressionada a tecla CONTROL para selecionar mais de uma. Usando a tecla SHIFT pode-se selecionar uma seqüência. Ao apertar o botão OK, elas irão para a lista de variáveis acompanhadas.

É possível exibir essa lista a qualquer momento, usando o menu Depurar|Variáveis, ou o botão respectivo da barra de ferramentas, ou ainda as teclas CONTROL e R ao mesmo tempo.

FIGURA 63 - A LISTA DE VARIÁVEIS ACOMPANHADAS



Sempre que a consulta alterar uma dessas variáveis, essa janela atualizar-se-á automaticamente. Para retirar uma variável da lista de acompanhamento, selecione-a com o mouse e pressione a tecla DELETE.

Para poder utilizar livremente as janelas de acompanhamento, é necessário que o projetista da base não tenha inserido uma proteção por senha.

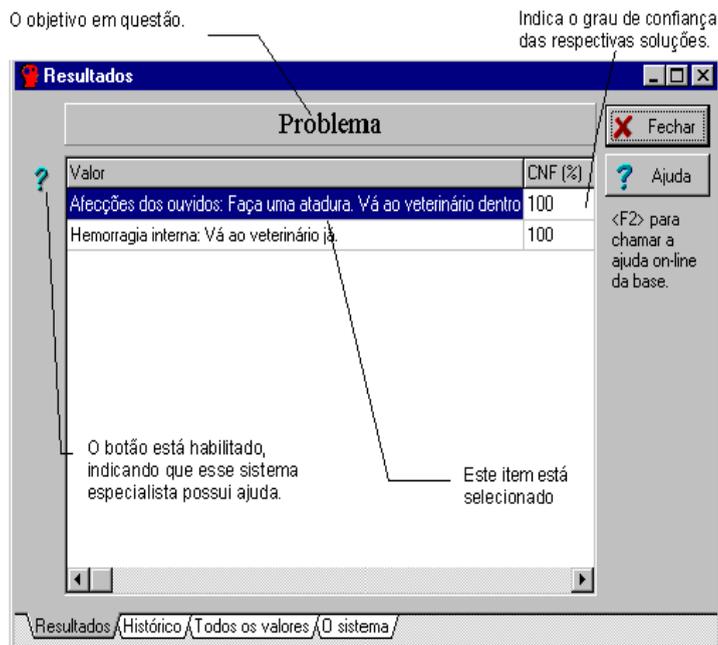
4.4.5 - Compreendendo os Resultados Atingidos

O sistema especialista divide a sua consulta em uma busca de objetivos. Ao final de cada busca, uma janela de resultados é apresentada. Ela se divide em quatro partes:

Resultados: são apresentados todos os valores atingidos pelo objetivo, com os respectivos graus de confiança. Caso o sistema não tenha chegado a nenhuma solução, ele indicará.

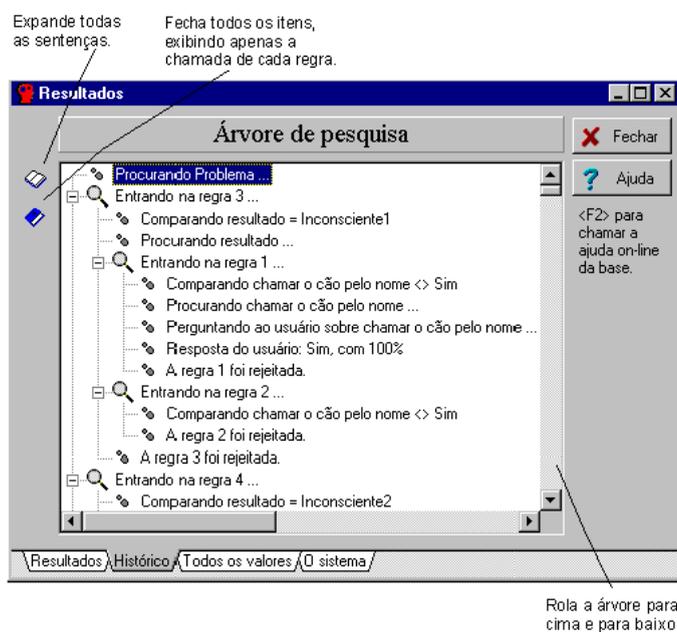
Se o pequeno botão localizado no canto superior esquerdo, estiver marcado com um ponto de interrogação e estiver habilitado, significa que o projetista da base também criou uma ajuda sobre o sistema especialista. Assim sendo, selecione no grid de qual solução você deseja maiores explicações (um leve retângulo envolve a seleção). Em seguida, pressione esse botão, e um texto explicativo sobre a solução aparecerá. Ou um clique duplo sobre a solução, ou ainda pressione F2 com a solução desejada selecionada. O botão Ajuda localizado à direita apenas mostra como se deve utilizar a janela de resultados.

FIGURA 64 - OS RESULTADOS ATINGIDOS



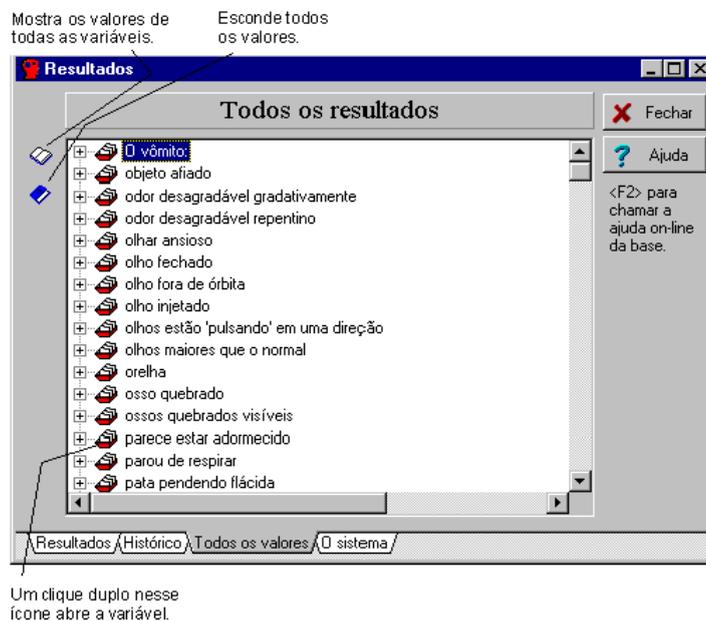
Histórico: exibe todo o caminho realizado pelo sistema especialista até atingir àquela(s) solução(ões). Esta página só estará disponível livremente se o projetista da base não houver protegido a base de conhecimento.

FIGURA 65 - A ÁRVORE DE PESQUISA



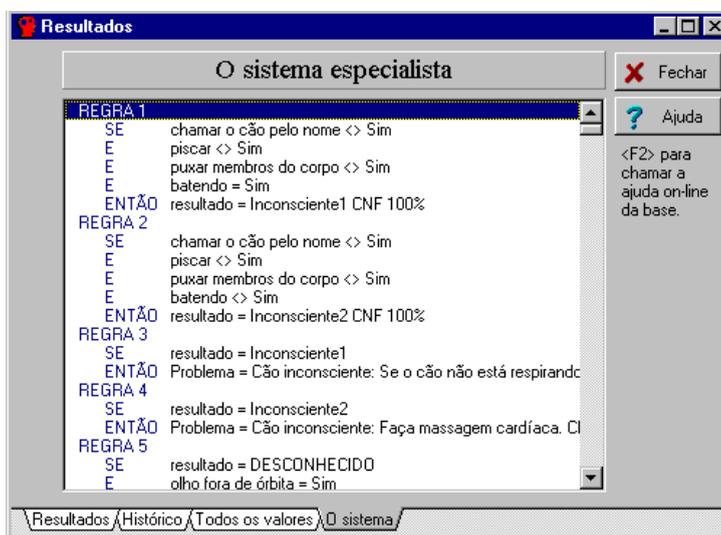
Todos os Valores: é uma generalização da primeira página. Exibe todos os valores de todas as variáveis. Também só estará disponível com permissão do projetista.

FIGURA 66 - TODOS OS VALORES ATINGIDOS PELO SISTEMA ESPECIALISTA



O Sistema: exibe as todas as regras do sistema especialista. Útil também para entender como o sistema chegou àquela(s) conclusão(ões), em conjunto com o histórico.

FIGURA 67 - A BASE DE CONHECIMENTO



Essa janela pode aparecer em uma consulta tantas vezes quanto for o número de objetivos a serem alcançados.

Pressione o botão Fechar para continuar a consulta.

4.5 - O DOMÍNIO DE CONHECIMENTO (SAÚDE DE CÃES)

Extraír conhecimentos de especialistas veterinários não é uma tarefa fácil, o principal motivo que leva os mesmos a dificultar o repasse de qualquer informação é o “crer” que sistemas especialistas podem substituí-los, porém, ao mesmo tempo não se dão conta que esses sistemas podem ajudá-lo a lembrar de casos não comuns, servindo como uma memória adicional. Conseqüentemente, a ajuda prestada por parte de dois veterinários foi muito superficial, sendo que o principal “alvo” de obtenção de informações foi o livro “Primeiros Socorros para Cães” de Bruce Fogle [30]. Além de descrever o que fazer quando ocorrem emergências, também indica se e quando é preciso consultar um veterinário.

O livro é dividido em quatro partes. A Parte 1 relata o que fazer primeiro no caso de uma emergência; como imobilizar o cão, checar seus sinais vitais, preservar sua vida, aplicar ataduras e torniquetes e transportá-lo com segurança.

A Parte 2 explica, passo a passo, o que fazer quando a crise imediata passar. Se você não está seguro de como agir, essa seção irá ajudá-lo a decidir se a assistência veterinária é necessária e, em caso positivo, com que urgência deve procurá-la. Situações de ameaça potencial à vida e condições que podem provocar dores não-aparentes são focalizadas com símbolos facilmente reconhecíveis.

A Parte 3 relata o que fazer quando a situação é, nitidamente, de emergência. As emergências estão listadas em ordem alfabética, de agressão a convulsões, de diabetes a patas inchadas e, finalmente, a vômitos. Para cada uma, há uma descrição de como aplicar os primeiros socorros e uma tabela para ajudá-lo a, algumas vezes, decidir sobre se é necessário procurar o veterinário o mais rápido possível, se se pode esperar até o final do dia ou o dia seguinte ou, ainda, se se deve simplesmente telefonar em busca de orientação ou para ter a segurança de apenas continuar com o tratamento doméstico.

A Parte 4 mostra como minimizar os riscos para o seu cão, como ministrar medicamentos e como preparar um *kit* de primeiros socorros.

V - DISCUSSÃO E CONCLUSÕES

5.1 - ANÁLISE DOS RESULTADOS

Os resultados obtidos com o sistema SEPRISCA foram os esperados, isto é, o sistema encontrou corretamente os problemas de acordo com as regras formuladas. Vale-se ressaltar que o sistema se propõe a examinar todas as possíveis situações de emergência que podem ocorrer com seu cão, indicando *se* e *quando* é preciso visitar o veterinário.

A obtenção de informações dos especialistas nesta área tornou-se um obstáculo, dificultando o processo de desenvolvimento da base de conhecimentos.

A respeito do tratamento da imprecisão com variáveis lingüísticas pode-se dizer que é mais próximo da linguagem utilizada pelo especialista, do que os sistemas dicotômicos que só permitem dois estados possíveis aos sintomas.

Mesmo sem resposta a determinadas perguntas, o sistema poderá levar a resultados, pelo fato de conter um valor especial, passível de ser incluído em todas as variáveis; o chamado “DESCONHECIDO”, que representa uma indeterminação total sobre as instanciações de uma variável, fazendo com que o processo prossiga sua execução no encontro de respostas para o problema.

5.2 - DISCUSSÃO

A interação engenheiro de conhecimento e o especialista médico veterinário, ainda não tem atingido o nível desejado. De fato as bases de conhecimento entre ambos são diferentes, os preconceitos de um lado do médico veterinário em acreditar que o engenheiro só faz raciocínios sobre valores exatos e do engenheiro em acreditar que o médico veterinário faz raciocínios não demonstrativos pela lógica, impedem a comunicação entre eles.

Nesse contexto tem-se que:

A dificuldade de comunicação com especialistas na área dificulta o desenvolvimento deste tipo de sistema, o que faz uma necessidade da área da Inteligência Artificial interagir mais com os veterinários.

Dependendo do caso a ser estudado e analisado, a resolução de problemas usando regras de produção é uma boa opção, dado que muitos SE, usando esta forma de representação de conhecimento, foram desenvolvidos em diferentes áreas: Medicina, Agronomia, Prospecção de Petróleo, Moléculas Químicas, Controle de Poluição.

Um bom sistema especialista não deve saber somente encontrar respostas, mas também fazer o usuário utilizar a resposta com fins práticos. Para isso, o Expert SINTA possibilita a inclusão de tópicos de ajuda para os valores possíveis de determinados atributos, associando-o a um arquivo no formato Ajuda do Windows a sua base.

Visando a extrema necessidade destes textos explicativos, no qual a idéia é ir ao tópico explicativo de uma resposta, e, a falta de conhecimento sobre montagem de arquivos de ajuda, levou-me a buscar recursos para o desenvolvimento do mesmo, tarefa a qual encontrei dificuldades devido a falta de clareza no conteúdo do material utilizado, bem como, na disposição das imagens que descrevem o procedimento a ser tomado à determinadas emergências.

5.3 - CONCLUSÕES

O esforço foi dirigido de modo a obter maiores conhecimentos na área de Inteligência Artificial, em relação a Sistemas Especialistas. Conhecendo na prática o desenvolvimento e funcionamento do mesmo. Demonstrando que é possível desenvolver sistemas que trabalhem com variáveis lingüísticas e não precisam ter valores exatos para chegar a resultados. Finalmente, pode-se concluir que:

A utilização de variáveis lingüísticas é mais realista do que um sistema dicotômico de sim ou não.

O sistema pode ser olhado de modo diferente pelos especialistas, posto que ele não tenta substituí-los, mas, pode servir como uma “memória adicional” e fazê-los lembrar de casos não comuns.

Os sistemas de produção são a forma de representação de conhecimento mais usada em IA. A causa disto reside no fato de ser natural ao humano usar o par “condição-ação” para raciocinar e decidir, modo pelo qual facilita o desenvolvimento do SE.

VI - ANEXOS

6.1 - ANEXO 01 - UMA SUCINTA DESCRIÇÃO DA BASE DE CONHECIMENTOS DO SISTEMA SEPRISCA

REGRA 1
SE chamar o cão pelo nome <> Sim
E piscar <> Sim
E puxar membros do corpo <> Sim
E batendo = Sim
ENTÃO resultado = Inconsciente1 CNF 100%

REGRA 2
SE chamar o cão pelo nome <> Sim
E piscar <> Sim
E puxar membros do corpo <> Sim
E batendo <> Sim
ENTÃO resultado = Inconsciente2 CNF 100%

REGRA 3
SE resultado = Inconsciente1
ENTÃO Problema = Cão inconsciente: Se o cão não está respirando, aplique respiração boca-a-boca. CNF 100%

REGRA 4
SE resultado = Inconsciente2
ENTÃO Problema = Cão inconsciente: Faça massagem cardíaca. CNF 100%

REGRA 5
SE resultado = DESCONHECIDO
E olho fora de órbita = Sim
OU superfície do globo ocular = Sim
OU sangue ou algum corpo estranho = Sim
ENTÃO Problema = Afecções dos olhos: Primeiros socorros. Vá ao veterinário já. CNF 100%

REGRA 6
SE resultado = DESCONHECIDO
E olho fora de órbita <> Sim
E superfície do globo ocular <> Sim
E sangue ou algum corpo estranho <> Sim
E olho fechado = Sim
OU mudança na capacidade visual = Sim
OU pupilas = Sim
OU superfície do olho = Sim
OU olho injetado = Sim
OU olhos maiores que o normal = Sim
ENTÃO Problema = Afecções dos olhos: Vá ao veterinário dentro de 12 horas. CNF 100%

REGRA 7
SE resultado = DESCONHECIDO
E olho fora de órbita <> Sim
E superfície do globo ocular <> Sim
E sangue ou algum corpo estranho <> Sim
E olho fechado <> Sim
E mudança na capacidade visual <> Sim
E pupilas <> Sim
E superfície do olho <> Sim
E olho injetado <> Sim
E olhos maiores que o normal <> Sim
E secreção verde ou amarela = Sim
E filhote que não abriu os olhos = Sim
OU Problema = Afecções dos olhos: Vá ao veterinário dentro de 24 horas. CNF 100%

REGRA 8
SE resultado = DESCONHECIDO
E olho fora de órbita <> Sim
E superfície do globo ocular <> Sim
E sangue ou algum corpo estranho <> Sim
E olho fechado <> Sim
E mudança na capacidade visual <> Sim
E pupilas <> Sim
E superfície do olho <> Sim
E olho injetado <> Sim
E olhos maiores que o normal <> Sim
E secreção verde ou amarela <> Sim
E filhote que não abriu os olhos <> Sim
E secreção clara e incolor = Sim
OU espirros e esfregar a cara = Sim
ENTÃO Problema = Afecções dos olhos: Tratamento doméstico. CNF 100%

REGRA 9
SE resultado = DESCONHECIDO
E orelha = Sangrando profundamente?
OU orelha = Inchada e quente?
ENTÃO Problema = Afecções dos ouvidos: Faça uma atadura. Vá ao veterinário dentro de 12 horas. CNF 100%

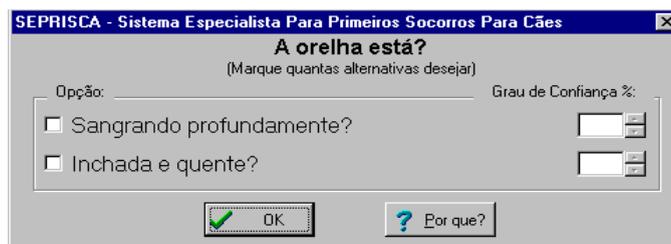
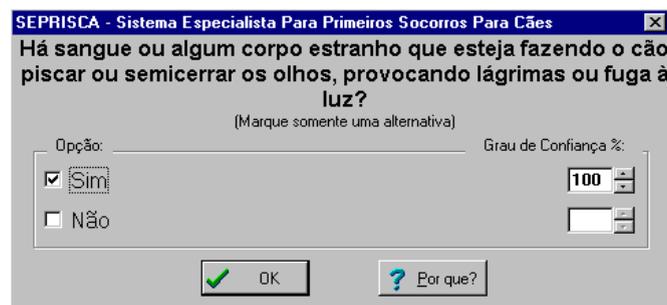
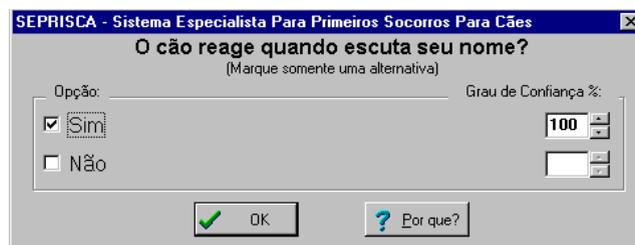
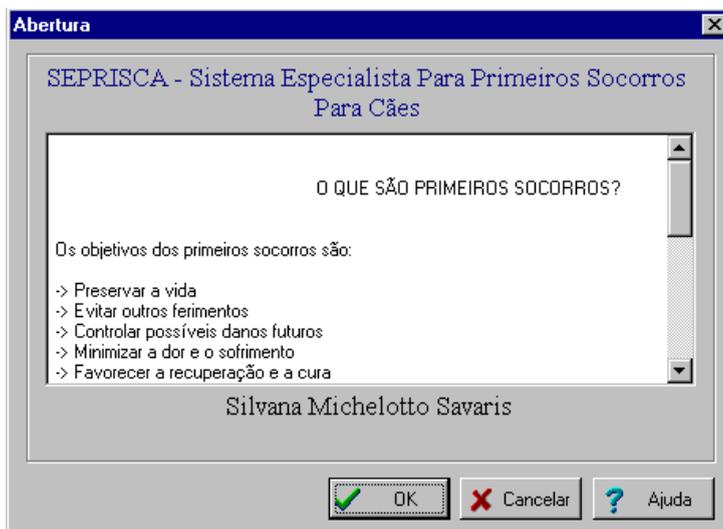
REGRA 10
SE resultado = DESCONHECIDO
E orelha = DESCONHECIDO
E O cão está(2 - Afecções dos ouvidos) = Sacudindo a cabeça?
OU O cão está(2 - Afecções dos ouvidos) = Com um lado da cabeça mais baixo que o outro?
ENTÃO Problema = Afecções dos ouvidos: Há um corpo estranho no canal do ouvido. Vá ao veterinário dentro de 24 horas. CNF 100%

REGRA 11	SE	resultado = DESCONHECIDO
	E	orelha = DESCONHECIDO
	E	O cão está:(2 - Afecções dos ouvidos) = DESCONHECIDO
	E	O cão está:(1 - Afecções dos ouvidos) = Perdendo a coordenação e equilíbrio?
	OU	O cão está:(1 - Afecções dos ouvidos) = Sem comer?
	OU	O cão está:(1 - Afecções dos ouvidos) = Vomitando?
	OU	O cão está:(1 - Afecções dos ouvidos) = Caindo ou andando em círculos?
	ENTÃO	Problema = Afecções dos ouvidos: A infecção atingiu o ouvido interno.Vá ao veterinário já. CNF 100%
REGRA 12	SE	resultado = DESCONHECIDO
	E	afito ou inquieto = Sim
	ENTÃO	Problema = Agressão e mudanças de comportamento: Vá ao veterinário já. CNF 100%
REGRA 13	SE	resultado = DESCONHECIDO
	E	afito ou inquieto <> Sim
	E	histérico ou em estado de pânico = Sim
	ENTÃO	Problema = Agressão e mudanças de comportamento: Vá ao veterinário já. CNF 100%
REGRA 14	SE	resultado = DESCONHECIDO
	E	afito ou inquieto <> Sim
	E	histérico ou em estado de pânico <> Sim
	E	se recuperou, mas não é o mesmo = Sim
	ENTÃO	Problema = Agressão e mudanças de comportamento: Telefone em busca de orientação. CNF 100%
REGRA 15	SE	resultado = DESCONHECIDO
	E	afito ou inquieto <> Sim
	E	histérico ou em estado de pânico <> Sim
	E	se recuperou, mas não é o mesmo <> Sim
	E	se recuperou e está agindo normalmente = Sim
	ENTÃO	Problema = Agressão e mudanças de comportamento: Telefone em busca de orientação. CNF 100%
REGRA 16	SE	resultado = DESCONHECIDO
	E	anzol preso a língua ou ao céu da boca = Sim
	OU	vários anzóis presos = Sim
	OU	assustado ou perigoso demais = Sim
	ENTÃO	Problema = Anzóis: Vá ao veterinário já. CNF 100%
REGRA 17	SE	resultado = DESCONHECIDO
	E	anzol preso a língua ou ao céu da boca <> Sim
	E	vários anzóis presos <> Sim
	E	assustado ou perigoso demais <> Sim
	E	anzol preso a língua, pata ou corpo = Sim
	ENTÃO	Problema = Anzóis: Primeiros socorros. CNF 100%
REGRA 18	SE	resultado = DESCONHECIDO
	E	O cão está apresentando algum dos seguintes sintomas:(Asfixia) = Bater a pata na boca?
	OU	O cão está apresentando algum dos seguintes sintomas:(Asfixia) = Aflição evidente?
	OU	O cão está apresentando algum dos seguintes sintomas:(Asfixia) = Sons de asfixia?
	OU	O cão está apresentando algum dos seguintes sintomas:(Asfixia) = Olhos saltados?
	OU	O cão está apresentando algum dos seguintes sintomas:(Asfixia) = Língua azulada?
	OU	O cão está apresentando algum dos seguintes sintomas:(Asfixia) = Agitação ou inconsciência?
	ENTÃO	Problema = Asfixia: Aplique primeiros socorros para asfixia imediatamente. CNF 100%
REGRA 19	SE	resultado = DESCONHECIDO
	E	O cão está apresentando algum dos seguintes sintomas:(Asfixia) = DESCONHECIDO
	E	Algum dos seguintes sintomas(Asfixia) = Bater a pata na boca?
	OU	Algum dos seguintes sintomas(Asfixia) = Sufocamento?
	OU	Algum dos seguintes sintomas(Asfixia) = Dificuldades respiratórias?
	OU	Algum dos seguintes sintomas(Asfixia) = Aflição branda?
	OU	Algum dos seguintes sintomas(Asfixia) = Esfregar a cara no cão?
	OU	Algum dos seguintes sintomas(Asfixia) = Hálito desagradável?
	ENTÃO	Problema = Asfixia: Procure por algum objeto preso na boca. CNF 100%
REGRA 20	SE	resultado = DESCONHECIDO
	E	o cão foi exposto à corrente elétrica = Está inconsciente?
	OU	o cão foi exposto à corrente elétrica = Está tendo convulsões?
	OU	o cão foi exposto à corrente elétrica = Está prostado?
	OU	o cão foi exposto à corrente elétrica = Está sem batimento cardíaco?
	OU	o cão foi exposto à corrente elétrica = Esvaziou a bexiga e os intestinos?
	ENTÃO	Problema = Choque elétrico: Faça ressuscitamento cardiopulmonar. Vá ao veterinário já. CNF 100%
REGRA 21	SE	resultado = DESCONHECIDO
	E	o cão foi exposto à corrente elétrica = DESCONHECIDO
	E	queimadura cujo centro é claro cercado por vermelhidão = Sim
	OU	respiração mais rápida ou mais lenta = Sim
	ENTÃO	Problema = Choque elétrico: Trate da queimadura. Vá ao veterinário no mesmo dia. CNF 100%
REGRA 22	SE	resultado = DESCONHECIDO
	E	Há claudicação e: = Dor, sofrimento, fraqueza súbita?
	OU	Há claudicação e: = Osso ou membro curvado de modo visivelmente anormal?
	OU	Há claudicação e: = O cão geme quando movimentado?
	OU	Há claudicação e: = Há registro de algum trauma? (Coice de cavalo, acidente de carro, etc.)
	OU	Há claudicação e: = O cão não move a cabeça ou o pescoço?
	OU	Há claudicação e: = Há hemorragia?
	OU	Há claudicação e: = Perda de controle de emissão de fezes ou urina?
	OU	Há claudicação e: = Pode-se sentir um osso tocando o outro?
	ENTÃO	Problema = Claudicação e dificuldades de movimentação: Vá ao veterinário já. CNF 100%
REGRA 23	SE	resultado = DESCONHECIDO
	E	Há claudicação e: = DESCONHECIDO
	E	junta quente, inchada ou dolorida = Sim
	OU	está muito tenso, especialmente nas costas = Sim
	ENTÃO	Problema = Claudicação e dificuldades de movimentação: Vá ao veterinário dentro de 12 horas. CNF 100%

- REGRA 24**
SE resultado = DESCONHECIDO
E Há claudicação e: = DESCONHECIDO
E junta quente, inchada ou dolorida <> Sim
E está muito tenso, especialmente nas costas <> Sim
E não consegue apoiar seu peso numa das pernas = Sim
OU o estado é normal, mas ele ainda está mancando após um repouso de 48 horas = Sim
OU inchaço numa das patas, piche, unha quebrada ou almofadas arranhadas = Sim
ENTÃO Problema = Claudicação e dificuldades de movimentação: Vá ao veterinário dentro de 24 horas. CNF 100%
- REGRA 25**
SE resultado = DESCONHECIDO
E O cão está:(Choque) = Com gengivas descoradas ou brancas?
OU O cão está:(Choque) = Respiração acelerada?
OU O cão está:(Choque) = Pulso fraco e rápido?
OU O cão está:(Choque) = Extremidades frias?
OU O cão está:(Choque) = Fraqueza generalizada?
ENTÃO Problema = Choque CNF 100%
- REGRA 26**
SE resultado = DESCONHECIDO
E O cão:(Coceira) = Recebeu medicamentos via oral ou por injeção?
OU O cão:(Coceira) = Foi picado por vespa, abelha, marimbondo ou formiga?
OU O cão:(Coceira) = Brincou com lagartas?
OU O cão:(Coceira) = Esteve em contato com urtigas?
ENTÃO Problema = Coceira: Aplique uma compressa. Telefone em busca de orientação. CNF 100%
- REGRA 27**
SE resultado = DESCONHECIDO
E parece estar adormecido = Sim
OU o cão respira = Sim
OU não reage a voz = Sim
OU não reage ao toque = Sim
ENTÃO Problema = Coma: Vá ao veterinário já. CNF 100%
- REGRA 28**
SE resultado = DESCONHECIDO
E O esforço para evacuar está associado a algum desses sintomas: = Dor?
OU O esforço para evacuar está associado a algum desses sintomas: = Vômito?
OU O esforço para evacuar está associado a algum desses sintomas: = Febre?
OU O esforço para evacuar está associado a algum desses sintomas: = Fezes semelhantes a fitas?
OU O esforço para evacuar está associado a algum desses sintomas: = Pequenas quantidades de fezes malcheirosas?
OU O esforço para evacuar está associado a algum desses sintomas: = Sangue?
OU O esforço para evacuar está associado a algum desses sintomas: = Perda de peso?
OU O esforço para evacuar está associado a algum desses sintomas: = Uma saliência ao lado do ânus?
OU O esforço para evacuar está associado a algum desses sintomas: = Fio, barbante ou outro material feito pelo homem pendendo do ânus?
ENTÃO Problema = Constipação: Vá ao veterinário dentro de 24 horas. CNF 100%
- REGRA 29**
SE resultado = DESCONHECIDO
E O esforço para evacuar está associado a algum desses sintomas: = DESCONHECIDO
E grama projetando-se do ânus = Sim
OU fezes presas ao pêlo ao redor do ânus = Sim
OU antecedentes de constipação ocasional = Sim
ENTÃO Problema = Constipação: Tratamento doméstico. CNF 100%
- REGRA 30**
SE resultado = DESCONHECIDO
E O cão está:(1 - Convulsões e ataques) = Deitado de lado, prostrado?
OU O cão está:(1 - Convulsões e ataques) = Contorcendo-se?
OU O cão está:(1 - Convulsões e ataques) = Arqueando as costas ou batendo as patas como se estivesse andando de bicicleta?
OU O cão está:(1 - Convulsões e ataques) = Urinando ou defecando?
OU O cão está:(1 - Convulsões e ataques) = Salivando?
OU O cão está:(1 - Convulsões e ataques) = Revirando os olhos?
ENTÃO Problema = Convulsões e Ataques: Convulsão generalizada. CNF 100%
- REGRA 31**
SE resultado = DESCONHECIDO
E O cão está:(1 - Convulsões e ataques) = DESCONHECIDO
E O cão está:(2 - Convulsões e ataques) = Fitando o vazio?
OU O cão está:(2 - Convulsões e ataques) = Andando em círculos?
OU O cão está:(2 - Convulsões e ataques) = Aparentemente cego?
OU O cão está:(2 - Convulsões e ataques) = Apanhando moscas inexistentes?
OU O cão está:(2 - Convulsões e ataques) = Contorcendo-se levemente?
OU O cão está:(2 - Convulsões e ataques) = Apresentando mudanças de comportamento repentinas, inesperadas e de curta duração?
ENTÃO Problema = Convulsões e Ataques: Convulsão leve. CNF 100%

6.2 - ANEXO 02 - UMA SUCINTA DEMONSTRAÇÃO DA EXECUÇÃO DO SISTEMA SEPRISCA.

Tela de Abertura



SEPRISCA - Sistema Especialista Para Primeiros Socorros Para Cães

O cão está?
(Marque quantas alternativas desejar)

Opção: _____ Grau de Confiança %:

Sacudindo a cabeça?

Com um lado da cabeça mais baixo que o outro?

OK

SEPRISCA - Sistema Especialista Para Primeiros Socorros Para Cães

O cão está?
(Marque quantas alternativas desejar)

Opção: _____ Grau de Confiança %:

Perdendo a coordenação e equilíbrio?

Sem comer?

Vomitando?

Caindo ou andando em círculos?

OK

Demais telas são apresentadas, neste meio, durante a execução do sistema, porém, as mesmas não serão demonstradas devido a grande quantidade.

SEPRISCA - Sistema Especialista Para Primeiros Socorros Para Cães

O sangramento provém de um ferimento provocado por faca, bala ou outro objeto perfurante?
(Marque somente uma alternativa)

Opção: _____ Grau de Confiança %:

Sim

Não

OK

SEPRISCA - Sistema Especialista Para Primeiros Socorros Para Cães

A pata está inchada e o cão esteve:
(Marque quantas alternativas desejar)

Opção: _____ Grau de Confiança %:

Ao ar livre, onde há insetos que picam?

OK

Segue a execução do sistema, apresentando outras telas.

A tela abaixo é a penúltima tela a ser apresentada

SEPRISCA - Sistema Especialista Para Primeiros Socorros Para Cães

O vômito é produtivo e:
(Marque quantas alternativas desejar)

Opção: _____ Grau de Confiança %:

Ocorreu menos do que três vezes?

Não está acompanhado de diarreia?

Não é acompanhado de outros problemas?

OK

Tela de Resultados

Resultados

Problema	
Valor	CNF (%)
Afecções dos olhos: Primeiros socorros. Vá ao veterinário já.	100
Hemorragia interna: Vá ao veterinário já.	100

<F2> para chamar a ajuda on-line da base.

Resultados / Histórico / Todos os valores / O sistema

VII - REFERÊNCIAS BIBLIOGRÁFICAS

- [1] ADLASSNIG, KLAUS-PETER. Fuzzy Set Theory in Medical Diagnosis. IEEE Trans. on Systems, Man and Cybernetics, v. 16, n. 2, p. 260-265, march/april, 1986.
- [2] ALMEIDA, M. A. F. Aprender Atividade Inteligente: E se esta Inteligência for Parcialmente Artificial? - Dissertação de Mestrado. Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Ciências da Computação, Florianópolis, setembro, 1999.
- [3] APPELBAUM, L.;RUSPINI, E. H. ARIES. An Approximate Reasoning Inference Engine. Approximate Reasoning in Expert Systems. Gupta et al. (eds.), Elsevier Science Publishers, Amsterdam, p. 745-755, 1985.
- [4] BARR, A. ; FEIGENBAUM E. A. The Handbook of Artificial Intelligence. Heuristech Press, 1981, v. 1.
- [5] BARRETO, J. M. Inteligência Artificial e Engenharia Biomédica: Casamento Perfeito ou Amantes Eternos?. XVII Congresso Brasileiro de Engenharia Biomédica (CBEB2000), Florianópolis, setembro, 2000.
- [6] BARRETO, J. M. Inteligência Artificial no Liminar do Século XXI. Edições, Florianópolis, 1997.
- [7] BARRETO, JORGE. A Topological Model for Qualitative Time. Proceedings: IASTED Artificial Intelligence Application and Neural Networks, Zurich, Switzerland, p. 16-21, june 25-27, 1990.

- [8] BARRETO, JORGE. Expert Systems in Biomedicine. Université Catholique de Louvain, Faculté de Médecine - Faculté des Sciences Appliquées, Certificat en Informatique Biomédicale, Bruxellas, Belgium, 1993.
- [9] BARRETO, JORGE. Lógica Nebulosa. Universidade Federal de Santa Catarina, Centro Tecnológico, Departamento de Engenharia Elétrica, Grupo de Pesquisas em Engenharia Biomédica, Florianópolis, 1995.
- [10] BITTENCOURT, G. Inteligência Artificial - Ferramentas e Teorias. UFSC, Florianópolis, 1998.
- [11] BRASIL, L. M. Proposta de Arquitetura para Sistema Especialista Híbrido e a Correspondente Metodologia de Aquisição de Conhecimento -Tese de Doutorado. Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Engenharia Elétrica, área Engenharia Biomédica, Florianópolis, fevereiro, 1999.
- [12] BRASIL, L. M.; AZEVEDO, F. M.; BARRETO, J. M. A Hybrid Expert System for the Diagnosis of Epileptic Crisis. Artificial Intelligence in Medicine. Elsevier, n. 585, p. 1-7, 2000.
- [13] BRASIL, L. M.; AZEVEDO, F. M.; BARRETO, J. M. Expert System Used Hybridism among Symbolic and Connecionist Paradigms, Fuzzy Logic and Genetic Algorithms. Proceedings of the International Symposium on Medical Informatics and Fuzzy Technology (MIF'99), Hanoi, Vietnam, p. 347-354, 1999.
- [14] BRASIL, L. M.; AZEVEDO, F. M.; BARRETO, J. M. Hybrid Expert System for Decision Support in the Medical Area. In: Medical and Biological Engineering and Computing, Viena, Áustria, v. 37, n. 2, p. 738-739, 1999.
- [15] CELLIER, FRANCOIS. Continuous System Modelling. Springer-Verlag, New York, 1991.
- [16] CHORAFAS, D. N. Sistemas Especialistas: Aplicações Comerciais. McGraw-Hill,

- São Paulo, 1988.
- [17] DA ROCHA FERNANDES, A. M. Sistema Especialista Difuso Aplicado ao Processo de Análise Química Qualitativa de Amostras de Minerais - Dissertação de Mestrado. Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Ciências da Computação, Florianópolis, fevereiro, 1996.
- [18] DAZZI, R. L. S. Sistemas Especialistas Conexionistas, Implementados por Redes Diretas e Bidirecionais - Dissertação de Mestrado. Universidade Federal de Santa Catarina, Florianópolis, setembro, 1999.
- [19] DAZZI, R. L. S. Sistemas Especialistas Conexionistas: Implementação por Redes Diretas e Bidirecionais. Rev. Alcanse, ISSN 1413-2591, Ano VIII, v. 1, p. 107-115, 2001.
- [20] DAZZI, R. L. S.; BARRETO, J. M. Sistemas Especialistas Conexionistas em Reumatologia. UFSC, Florianópolis.
- [21] DE AZEVEDO, F. M. Contribution to the Study of Neural Networks in Dynamical Expert Systems - PhD Thesis. Institut d'Informatique, FUNDP, Namur, Bélgica, 1993.
- [22] DE NEYER, MARC. Contribution to Fuzzy Controller Analysis and Study of Several Fuzzy Control Systems - PhD Thesis (Preliminary Version). Université Catholique de Louvain, Bruxelles, Belgium, 1996.
- [23] DUBOIS, DIDIER; PRADE, HENRI. Combination and Propagation of Uncertainty with Belief Functions. Proceedings: IJCAI '85, p. 111-113, 1985.
- [24] DUBOIS, DIDIER; PRADE, HENRI. Fuzzy Sets and Systems Theory and Applications. Academic Press, New York, 1980.
- [25] FALQUETO, J.; LIMA, W. C.; BORGES, P. S. S.; et al. The Measurement of Artificial Intelligence: An IQ for Machines? Proceedings of the IASTED

- International Conference on Modelling, Simulation and Control, Innsbruck, Áustria, v. 1, p. 409-413, february 19-22, 2001. Acta Press, Anaheim, Calgary, Zurich, ISBN 0-88986-316-4, ISSN 1025-8973.
- [26] FARRENY, H. Les Systemes Experts: Principes et Exemples. CEPADUES-Editions, Toulouse, 1985.
- [27] FEIGENBAUM, E. A.; McCORDUCK, P. The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World. Michel Joseph ed., 1983.
- [28] FIESCHI, MARIUS. Intelligence Artificielle em Médecine des Systèmes Experts. MASSON, Paris, 1984.
- [29] FISCHLER, M. A. ; FIRSCHEIN, O. Intelligence & Computer: The Central Role of Representation. Al Expert, december, 1986.
- [30] FOGLE, BRUCE. Primeiros Socorros para Cães. Nobel, São Paulo, 1996.
- [31] FORSYTH, R. Expert Systems: Principles and Case Studies. Chapman and Hall Ltd., 1984.
- [32] GALLANT, S.I. Neural Network Learning and Expert Systems. The MIT Press, 1993.
- [33] GIARRATANO, JOSEPH. Experts Systems Principles and Programming. PWS-KENT Publishing Company, Boston, 1989.
- [34] GORDON, JEAN; SHORTLIFFE, EDWARD. The Dempster-Shafer Theory of Evidence. Rule-Based Expert Systems. B. Buchanan, E. Shortliffe (eds.), p. 272-292, 1985.
- [35] HARMON, Paul; KING, David. Sistemas Especialistas: A Inteligência Artificial Chega ao Mercado. Campus, Rio de Janeiro, 1988.
- [36] HARTMUT KOPPITKE, BRUNO; et al. Um Sistema Inteligente de Apoio

- à Decisão Estratégica Baseado em Inferência Probabilística. A ser publicado em: Anais do VII Congresso Nacional de Investigação Operacional, Aveiro, Portugal, 1-3 Abril, 1996.
- [37] HAYES-ROTH, F.; WATERMAN, D.A.; LENAT, D.B. Building Expert System. Addison-Wesley Publishing Company, Inc., 1983.
- [38] HECKERMAN DAVID; WELLMAN MICHAEL P. Bayesian Networks. Communications of the ACM, v. 38, n. 3, p. 27-30, march, 1995.
- [39] HEWITT, C. E. Description and Theoretical Analysis (Using Schemata) of Planner Technical Report. Massachusetts Institute of Technology, Technical Report TR-258, AI Laboratory, Cambridge, MA, USA, 1972.
- [40] <http://n27.udesc.br/demo/trabalhos/alunos/mc/what.html>. Data da consulta: 04 de agosto 2001.
- [41] <http://penta2.ufrgs.br/gr952/trab1/geren2.html>. Data da consulta: 04 de agosto 2001.
- [42] <http://www.din.uem.br/ia/intelige/especialistas/especialistas/eficacia.htm>. Data da consulta: 04 de agosto 2001.
- [43] <http://www.din.uem.br/ia/intelige/especialistas/especialistas/introdu.html>. Data da consulta: 04 de agosto 2001.
- [44] HUDSON, D.; COHEN, M.; DEEDWANIA, P. EMERGE. An Expert System for Chest Pain Analysis. Approximate Reasoning in Expert Systems. Gupta et al. (eds.), Elsevier Science Publishers, Amsterdam, p. 705-718, 1985.
- [45] KLIR, GEORGE; YUAN, B. Fuzzy Sets and Fuzzy Logic. Prentice-Hall, New Jersey, 1995.
- [46] LIMA, W. C.; BARRETO, J. M. Inteligência Artificial. Ciência Hoje, v. 7, n. 38, p. 50-56, 1987.

- [47] LINARES, K. S. C. Sistema Especialista Nebuloso para Diagnóstico Médico - Dissertação de Mestrado. Universidade Federal de Santa Catarina, Departamento de Engenharia Elétrica, Florianópolis, 1991.
- [48] LINARES, K. S. C.; BARRETO, J. M.; NASSAR, S. M. Fuzzy Analogical Reasoning for Medical Diagnosis – Paper AFO1206. In: 3rd Internet World Congress on Biomedical Sciences, Riken, page 6 pages, 10 novembro, 1996.
- [49] LINARES, K. S. C.; BARRETO, J. M.; NASSAR, S. M. Raciocínio por Analogia Fuzzy para Diagnóstico Médico: Taxa de Aprendizado por Base de Dados. Fórum de Ciências e Tecnologia em Saúde, Campos de Jordão, p. 623-624, 13-17 outubro, 1996.
- [50] McCARTHY, J. Circumscription - A Form of Non-Monotonic Reasoning. Artificial Intelligence, v. 13, n. 1-2, p. 27-29, 1980.
- [51] MENDEL, JERRY M. Fuzzy Logic Systems for Engineering: A Tutorial. Proceedings of the IEEE, v. 83, n. 3, p. 345-377, march, 1995.
- [52] MIN, L. S. Sistema Baseado em Conhecimento para Detecção e Classificação de Crises Epilépticas. Universidade Federal de Santa Catarina, Programa de Medicina Interna do Hospital Universitário, Florianópolis, novembro, 1994.
- [53] MINSKY, M. A Framework for Representing knowledge. In: The Psychology of Computer Vision. Patrick H. Winston, McGraw-Hill, New York, 1975.
- [54] MONTELLO, M. V. Sistema Especialista para Predições e Complicações Cardiovasculares Integrado a um Sistema de Controle de Pacientes Portadores de Diabetes Mellitus - Dissertação de Mestrado. Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Engenharia Elétrica, área Engenharia Biomédica, Florianópolis, fevereiro, 1999.
- [55] OSHITA, SHUZO; NAKAKIMURA, KAZUHIKO; SAKABE, TAKEFUMI.

- Hypertension Control During Anesthesia. IEEE Magazine in Medicine and Biology, p. 667-670, november/december, 1994.
- [56] PERL, JUDEA. Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference. Morgan Kaufmann Publishers, San Mateo, CA, 1988.
- [57] POLYA, GEORGE. Mathematics and Plausible Reasoning Introduction and Analogy in Mathematics. Princenton University Press, New Jersey, 1990.
- [58] POLYA, GEORGE. Mathematics and Plausible Reasoning Patterns of Plausible Inference. Princenton University Press, New Jersey, 1990.
- [59] QUIBEL J. MARI R. Les Systèmes Experts dans l'Entreprise: Pourquoi, Comment? Les Éditions d'Organisation, Paris, 1989.
- [60] RABUSKE, R. A. Inteligência Artificial. UFSC, Florianópolis, 1995.
- [61] RICH, Elaine. Inteligência Artificial. McGraw-Hill Ltda, São Paulo, 1988.
- [62] RICH, ELAINE; KNIGHT, KEVIN. Inteligência Artificial. MAKRO Books do Brasil, São Paulo, 1993.
- [63] SANCHEZ, E. Medical Diagnosis and Composite Fuzzy Relations. Advances in Fuzzy Set Theory and Applications. M. M. Gupta, R. K. Radage, R. R. Yager (eds.), North Holland, New York, p. 421-433, 1979.
- [64] SCHANK, R. C. Dynamic Memory: A Theory of Reminding and Learning in Computers and People. Cambridge University Press, New York, 1977.
- [65] SCHANK, R. C. Identification of Conceptualizations Underlying Natural Language. In: Computer Models of Thought and Language, R. C. Schank and K. M. Colby (eds.), Freeman, San Francisco, 1973.
- [66] SHAFER, GLENN. A Mathematical Theory of Evidence. Princeton University Press, 1976.
- [67] SHORTLIFFE, E. H. MYCIN. Computer-Based Medical Consultations. Elsevier

- Scientific, New York, 1976.
- [68] SHORTLIFFE, E. H.; LAWRENCE, M. F.; HENRION, MAX; et al. Reasoning under Uncertainty in Medical Decision-Support. Disponível na internet em: <http://www.camis.stanford.edu/Reasonin.htm>, 1996.
- [69] STALLMAN, R. M.; SUSSMAN, G. J. Forward Reasoning and Dependency-Directed Back-Tracking in a System for Computer-Aided Circuit Analysis. Artificial Intelligence, v. 9, p. 2, 1977.
- [70] STEIN, C. E. Sistema Especialista Probabilístico: Base de Conhecimento Dinâmica - Dissertação de Mestrado. Universidade Federal de Santa Catarina, Florianópolis, agosto, 2000.
- [71] UMEYAMA, SINJI. The Complementary Process of Fuzzy Medical Diagnosis and Its Proprieties. Information Sciences, v. 38, p. 229-242, 1986.
- [72] WHALEN, Th.; SCHOTT, B. Goal-Directed Approximate Reasoning in a Fuzzy Production System. Approximate Reasoning in Expert Systems. Gupta et al. (eds.), Elsevier Science Publishers, Amsterdam, p. 505-518, 1985.
- [73] WINOGRAD, T. Language as a Cognitive Process-Syntax. Addison-Wesley Publishing Company, 1983.
- [74] WINSTON, P. H. Artificial Intelligence. Addison Wesley, 1979.
- [75] ZADEH, LOFTHI. Fuzzy Sets as A Basis for a Theory of Possibility. Fuzzy Sets and Systems, v. 1, p. 3-28, 1978.
- [76] ZADEH, LOFTHI. Fuzzy Sets. Information and Control, v. 8, p. 338-353, 1965.
- [77] ZIMMERMANN, HANS J. Fuzzy Sets, Decision Making and Expert Systems. Kluwer Academic Publishers, Boston, 1991.