

**UNIVERSIDADE FEDERAL DE SANTA CATARINA**  
Centro Tecnológico  
Curso de Pós-Graduação em Ciência da Computação

**Raul Domingos Valle Monteiro**

***UMA PROPOSTA DE APRENDIZAGEM INTERATIVA DA  
LÓGICA UTILIZANDO SIMULADORES ARTIFICIAIS  
INTELIGENTES***

Dissertação de Mestrado apresentada ao Programa  
de Pós-Graduação em Ciência da Computação

**Orientador: Prof. Dr. Jorge Muniz Barreto**

Florianópolis - SC, outubro de 2002

# DEDICATÓRIA

Aos meus filhos Raul Vitor, minha maior obra, e Elisa Catarina, meu maior presente, razões da minha existência. Dedico a vocês dois, toda a minha vida, recheada de vitórias e derrotas, decepções e alegrias, mas com o essencial: amor, honestidade e sucesso. Herdarão meus exemplos, cumprindo a lei natural da vida, o que me faz ser muito mais feliz.

# AGRADECIMENTOS

- A Deus, fonte de vida e verdade?

*“Se alguém diz? – eu amo a Deus e aborrece o seu irmão, é mentiroso. Pois quem não ama o seu irmão, ao qual viu, como pode amar a Deus a quem não viu” – Jesus Cristo.*

*Em nossas aflições o Pai é invocado.*

*Nas alegrias, é adorado.*

*Na noite tempestuosa é sempre esperado com ânsia.*

*No dia festivo é reverenciado solenemente.*

*Louvido pelos filhos reconhecidos e olvidado pelos ingratos, o Pai espalha as bênçãos de sua bondade infinita entre bons e maus, justos e injustos.*

*Ensina o verme a rastejar, o arbusto a desenvolver-se e o homem a raciocinar.*

*Ninguém duvide, porém, quanto à expectativa do Supremo Senhor a nosso respeito.*

*Jamais chegaremos, contudo, a semelhante condição, simplesmente através dos mil modos de coloração brilhante dos nossos sentimentos e raciocínios.*

*Nossos ideais superiores são imprescindíveis, e, no fundo, assemelham-se às flores mais belas e perfumadas da árvore.*

*Nossa cultura é, sem dúvida, indispensável, e, em essência, constitui a robustez do tronco respeitável.*

*Nossas aspirações elevadas são preciosas e necessárias e representam as folhas vivas e promissoras.*

*Assim também ocorre nos domínios da alma. Somente é possível glorificar o Pai quando nos abrimos aos seus decretos de amor universal, produzindo para o bem eterno.*

*Que nossa atividade, dentro da vida, produza muitos frutos de paz e sabedoria, amor e esperança, fé e alegria, justiça e misericórdia, em trabalho pessoal digno e constante, porquanto, assim o Pai será por nós glorificado e só nessa condição seremos discípulos do Mestre crucificado e redivivo.*

- Ao Prof. Dr. Jorge Muniz Barreto, que com sua sabedoria profunda, muito me orgulhou ao fazer parte do meu trabalho e também da minha vida.
- Aos queridos Professores que participaram do Programa de Mestrado, abrindo novos horizontes em nossas vidas: Custódio, Ricardo, Bosco Sobral, Daniela, Westphal, Zancanella, Paulo Sérgio e Fábio.
- Ao Prof. Dr. José Leônidas Olinquevitch, pela sua liderança, competência e confiança, o que sempre me encorajou muito em minha carreira profissional.
- Ao Prof. MsC. Carlos Shinoda, pelas demonstrações de serenidade, equilíbrio e organização, o que sempre me orientou pelos caminhos do sucesso.
- Ao Dr. Edu Arruda Junior pela confiança, segurança e sucesso que sempre alcançou, dedico com muito orgulho todo o meu respeito e admiração.
- A minha avó Catarina Valle, que ao se identificar e se afinar comigo desde minha infância, muito influenciou na minha formação com seus exemplos maravilhosos de conduta e humildade.
- Aos tios Dimas, Raul, Nery e Vary. Vocês têm o sobrinho mais orgulhoso deste mundo. Sou filho de vocês também. Pelos seus exemplos, pelas suas vidas e pelo amor à grande família, sempre demonstrado nos carinhos e atenção dispensados.
- Ao tio Pedro Valle, que apesar de a vida nunca ter permitido a nossa convivência próxima, reconheço e agradeço sua participação decisiva desde o início de todo o processo que desencadeou meu sucesso profissional.
- A minha namorada Lígia, companheira de todas as horas, sempre ao meu lado, em todos os momentos, com alegria, fé e consistência.
- Aos colegas de mestrado, que sempre participativos e unidos muito me empolgaram e colaboraram para o nosso sucesso.
- Aos meus pais Alfredo, homem honesto e perseverante e Lenice, exemplo de equilíbrio e amor.

# Sumário

<b>Lista de figuras.....</b>	<b>7</b>
<b>Lista de tabelas.....</b>	<b>8</b>
<b>Tabela de siglas e símbolos.....</b>	<b>9</b>
<b>Resumo.....</b>	<b>10</b>
<b>Abstract.....</b>	<b>11</b>
<b>1. Introdução.....</b>	<b>12</b>
1.1. Motivação.....	12
1.2. Objetivos.....	13
1.2.1. Objetivo geral.....	13
1.2.2. Objetivos específicos.....	13
1.3. Organização do trabalho.....	13
<b>2. Conceitos de Inteligência artificial.....</b>	<b>15</b>
2.1. IA na linha do tempo.....	16
2.2. Atividades psíquicas.....	29
2.3. Ferramentas utilizadas em IA.....	31
2.3.1. Sistemas de produção .....	31
2.3.2. Lógica nebulosa.....	31
2.4. Aplicações típicas em IA.....	34
2.4.1. Jogos.....	34
2.4.1.1. Grafos de jogos.....	36
2.4.1.2. O jogo da velha.....	36
2.4.2. Prova automática de teoremas.....	38
2.4.3. Sistemas especialistas.....	38
2.4.4. Compreensão da linguagem natural.....	41
2.4.5. Representação de incertezas.....	44
2.4.6. Percepção.....	48
2.4.7. Robótica.....	49
<b>3. Instrumentação, controle e automação.....</b>	<b>52</b>
3.1. Conceitos fundamentais.....	52

3.2. Problemas de controle.....	52
<b>4. Lógica e dedução.....</b>	<b>55</b>
4.1. Sistemas discretos.....	55
4.2. Cálculo sentencial.....	57
4.3. Cálculo de predicados.....	61
<b>5. Representação de conhecimento.....</b>	<b>62</b>
5.1. Representações Lógicas.....	62
5.2. Representações estruturadas.....	63
5.3. Métodos de busca.....	64
<b>6. Sistemas de informação inteligentes.....</b>	<b>66</b>
6.1. A engenharia de conhecimento.....	66
6.2. A extração de conhecimento.....	69
<b>7. O mapeamento cognitivo cerebral.....</b>	<b>72</b>
7.1. Mind Mapper 3.2.....	75
7.2. Mind Mapper Jr.....	77
<b>8. Simulador artificial para avaliação da aprendizagem da lógica.....</b>	<b>78</b>
8.1. A identidade de um software educativo.....	78
8.2. Aspectos pedagógicos do processo.....	79
8.3. Aspectos técnicos do processo.....	80
8.4. Processos metodológicos do SE proposto.....	81
8.5. Descrição da ferramenta principal.....	82
8.6. Definições e procedimentos da aplicação.....	85
8.6.1. Tela de apresentação: Proposta e Teoremas.....	85
8.6.2. Excitação: Combinação das entradas.....	86
8.6.3. A intervenção do aprendiz.....	86
8.6.4. Definição dos comandos – Células.....	87
8.6.5. Avaliação dos resultados.....	89
<b>9. Conclusão.....</b>	<b>91</b>
<b>Referências bibliográficas.....</b>	<b>92</b>

## LISTA DE FIGURAS

2.1. Fotos de Minsky, Simon, Newell e McCarthy.....	18
2.2. Grafo para o jogo da velha.....	37
3.1. Sistema de controle de nível de líquido em um tanque “il.color..”.....	53
4.1. Representação de um sistema discreto.....	55
4.2. Amostra do sinal $W(t)$ em instantes diversos.....	56
4.3. Conectivos sentenciais.....	57
4.4. Tabela verdade dos conectivos combinados.....	58
4.5. Tabela verdade dos conectivos com base em uma veracidade.....	58
4.6. Algumas expressões tautológicas.....	59
4.7. Legitimidade da dedução do problema “Tração excessiva”.....	60
5.1. Rede semântica representando equipamentos eletrônicos.....	63
5.2. Alinhamentos possíveis para os jogadores (O) e (X).....	64
6.1. Estrutura convencional de Sistema Especialista “il.color..”.....	66
6.2. Processo completo de extração de conhecimento em BD “il.color..”.....	70
6.3. Comparativo SI tradicional e Data mining.....	71
7.1. Organização funcional do Mind Mapper “il.color..”.....	73
7.2. Exemplo de um Mind Mapper utilizado em aprendizagem.....	74
7.3. Tela inicial de apresentação do Mind Mapper 3.2 “il.color..”.....	75
7.4. Disposição radial dos “Branches” e dos “Topics” “il.color..”.....	76
7.5. Disposição radial dos “Branches” e dos “Topics” na própria tela.....	76
7.6. Tela inicial do Mind Mapper Jr. “il.color..”.....	77
8.1. Esquema metodológico do SE proposto.....	82
8.2. Ícone do Mind Manager na área de trabalho do Windows.....	83
8.3. Esquema funcional do Mind Manager “il.color..”.....	83
8.4. Relações estabelecidas com o MS-Office “il.color..”.....	84
8.5. Estrutura do trabalho de avaliação da aprendizagem da lógica “il.color..”.....	85
8.6. Tela de apresentação do sistema em Power point.....	85

8.7. Tabela combinatória das excitações do sistema em Power point	86
“il.color..” .....	
8.8. Sistema simulador apresentado em Excel	87
“il.color..” .....	



# TABELA DE SIGLAS E SÍMBOLOS

## SIGLAS

<b>AA</b>	Agente Autônomo
<b>ALIFE</b>	Artificial Life ou Vida Artificial
<b>BD</b>	Base de Dados
<b>CN</b>	Conjuntos nebulosos
<b>DCP</b>	Dicionário de conceitos primitivos
<b>DF</b>	Dicionário de frases
<b>DT</b>	Dicionário de termos
<b>IA</b>	Inteligência Artificial
<b>IBM</b>	Empresa International Business Machines Corporation
<b>LF</b>	Linguagem formal
<b>LISP</b>	Linguagem de programação: List Programming
<b>LN</b>	Linguagem natural
<b>LN</b>	Lógica nebulosa
<b>RNA</b>	Redes Neurais Artificiais
<b>SE</b>	Sistemas Especialistas
<b>SI</b>	Sistemas de Informação
<b>SII</b>	Sistemas de Informação inteligentes

## SÍMBOLOS

$\forall$	Quantizador lógico: para todo
$\exists$	Quantizador lógico: existe
$\wedge$	Conectivo “E”
$\vee$	Conectivo “OU”
$\rightarrow$	Conectivo “Implica”
$\leftrightarrow$	Conectivo “Equivale”
$\neg$	Conectivo “Não”

## RESUMO

A computação, um dos mais variados e complexos processos produzidos e relacionados à atividade humana é um fenômeno global, irreversível e essencial. Esse processo pode trazer muitos resultados significativos em vários segmentos das atividades humanas.

Os resultados podem ser satisfatórios a partir de novas tecnologias e tornam os recursos otimizados e significativos dentro de suas atividades. Este trabalho aborda uma utilização prática para avaliar aprendizagem da lógica via Inteligência Artificial, seus efeitos e sua importância em nossa vida cotidiana.

Os estudos sobre processos de aprendizagem, abordados tradicionalmente sob o prisma apenas das condições internas da mente humana, passam a ter uma ótica diferenciada quando se utilizam da tecnologia para “tentar” simular o que realmente ocorre durante o processo de assimilação de novos conhecimentos, considerando todas, ou quase todas, as variantes envolvidas.

Serão descritas ferramentas utilizadas em Inteligência Artificial e uma metodologia própria que utiliza software específico desenvolvido e alguns relatos de experiências com resultados que certamente ainda não serão considerados suficientes para uma conclusão definitiva, mas serão, significativamente, validados pela metodologia de avaliação de aprendizagem apresentada, o que pode representar indicadores fortes para futuras aplicações.

# ABSTRACT

The computing of most varied and complex processes, products and services related to the human activity is a global, irreversible and essential phenomenon.

This process has many and incredible advantages. However, there is another aspect. Sometimes those effects are not beneficial, generic or similar, among the users of this technologies.

Many persons do not obtain satisfactory development from this newest technologies and can't use them in such way they are really useful and valuable to their activities. This work relates about the use of Artificial Intelligence for teaching in logic and its positive and negative effects the proposal also includes an approach about the impact on our lives.

The learning processes, in a traditional mind condition view, have now another different concern when the technology is on to "try" to simulate what actually occurs during the acquirement process for new knowledges by considering almost all the influence variables.

We show tools used from the Artificial Intelligence and we describe a new methodology that will become available through developed specific softwares. Some results of experiences are presented. Those results are not enough to consider a complete research but they are useful to validate this methodology and to become available for future applications.

# 1

## INTRODUÇÃO

*“Vulgar é o ler, raro o refletir. O saber não está na ciência alheia, que se absorve, mas, principalmente, nas idéias próprias, que se geram dos conhecimentos absorvidos, mediante a transmutação por que passam, no espírito que os assimila”.*

*Rui Barbosa – USP – 1921*

### 1.1. Motivação

Este trabalho trata da avaliação da aprendizagem no contexto do ensino de lógica formal, especificamente da lógica de Boole, utilizando simulação com Inteligência Artificial (IA).

A (IA) é um ramo da computação que foi constituído a partir de idéias filosóficas, científicas e tecnológicas herdadas de outras ciências, dentre elas, fundamentalmente, a Lógica.

Modelos implementados e teorias fundamentais que demonstram diretamente a ação inteligente, utilizando o computador como ferramenta, buscam analisar as capacidades cognitivas do ser humano e, mais fascinante ainda, desvendam mistérios da mente humana quando se deparam com outras incógnitas apresentadas durante o processo em que atuam.

Associada a um grande número de atividades nas mais diversas áreas de conhecimento, a IA tem registrado estratégias diversas para a solução de problemas, no desenvolvimento de algoritmos, métodos de processamento e apoio para a aprendizagem de temáticas características objetivas. Uma estratégia de avaliação de aprendizagem da lógica matemática proposta neste trabalho intenciona medir o que realmente se absorveu em conhecimentos lógicos, considerando o perfil do aprendiz o de um homem comum, dotado de capacidade intelectual.

## **1.2. Objetivos**

### **1.2.1. Objetivo geral**

Este trabalho tem como objetivo geral apresentar uma proposta alternativa em sistema especialista baseado na manipulação simbólica da representação de conhecimentos usando regras derivadas de domínio, envolvendo o raciocínio e a aprendizagem humana nas diversas áreas de aplicação, especialmente no aprendizado lógico-matemático.

Assim, sempre que se mencionar sistema de conhecimento, supõe-se que este conhecimento será manipulado simbolicamente utilizando métodos formais de reescrita. Que fique claro que não será tratada a representação de conhecimentos por valores de conexões sinápticas, representação esta que leva à abordagem conexionista, nem a valores de dados como os tratados em algoritmos genéticos, nem em abordagem similar.

### **1.2.2. Objetivos específicos**

Para que seja alcançado o objetivo geral, deverão ser alcançados alguns objetivos específicos:

- Especificar modelos lógicos que possam ser utilizados nas propostas de aprendizagem, incluindo modelos de automação e controle;
- Apresentar o comportamento do Sistema Especialista nas situações em que se apresenta como instrumento fundamental nos processos de avaliação da aprendizagem da lógica.

## **1.3. Organização do Trabalho**

O presente trabalho propõe uma estrutura pertinente e trata de uma sequência lógica própria e indicadora clara da sua proposta fundamental, assim dividida:

O Capítulo 1 (Introdução), que é este que está sendo lido no momento, apresenta a motivação para o trabalho e seus objetivos fundamentais, expondo com clareza e objetividade a fundamentação da proposta.

O Capítulo 2 (Conceitos de Inteligência Artificial) oferece uma visão geral sobre Inteligência Artificial, apresentando algumas das habilidades que o ser humano possui em termos de atividades psíquicas e listando algumas das ferramentas disponíveis no campo do ensino e ilustrando historicamente as principais áreas de aplicação das teorias da cognição..

O Capítulo 3 (instrumentação, controle e automação) apresenta os principais conceitos utilizados na teoria e na prática de controle e automação de sistemas dinâmicos, bem como descreve instrumentos que serão de muita importância no rigor da apresentação e comprovação da teoria proposta.

O Capítulo 4 (Lógica e dedução) mostra os fundamentos da lógica necessários ao estudo dos mecanismos de inferência empregados nos sistemas baseados em conhecimento. Os conceitos essenciais de lógica são fundamentalmente ligados aos manipuladores de bases de conhecimento.

O Capítulo 5 (Representação de conhecimento) discute formas de se representar conhecimentos, particularmente as que podem ser facilmente utilizadas em aprendizado.

O Capítulo 6 (Sistemas de Informação Inteligentes) especifica muitas funções dos Sistemas Inteligentes de Informação, cuidando das particularidades que podem ser relevantes na pesquisa implementada. Os SII's têm múltiplas aplicações e inferem e detêm princípios próprios, não dispensando desta forma, as especificidades da matéria que propõem.

O Capítulo 7 (O mapeamento cognitivo cerebral) trata especificamente da aprendizagem com base nos preceitos dos Capítulos anteriores, tentando relacionar a Engenharia de Conhecimento nas aplicações e no tratamento da funções lógicas ocorridas durante o aprendizado, incluindo as propostas objetivas do Mapeamento mental disponíveis como os software's Mind Mapper 3.2, Mind Mapper Jr, etc.

O Capítulo 8 (Simulador artificial para avaliação da aprendizagem da lógica) propõe sistemas automatizados de avaliação de aprendizagem que podem trazer significativos resultados na proposta de aprendizagem da lógica de Teoremas de Boole, implementando através do Software Mind Manager, interagindo com recursos do MS-Office como proposta final de avaliação de resultados obtidos.

# 2

## CONCEITOS DE INTELIGÊNCIA ARTIFICIAL

O conceito de “Vida Artificial” já ocupa grande parte das aplicações e pesquisas promovidas pelo homem, configurando desta forma, modelos tecnológicos emergentes em novas aplicações para implementação de modelos de gerenciamento das mais diversas atividades humanas. Seu conceito pode ser simplesmente definido como “Vida feita pelo homem e não pela natureza”. *Artificial Life*, ou simplesmente ALIFE, como é chamada, tem por premissa básica estudar a vida natural, tentando reproduzi-la por meios artificiais e, ao contrário da biologia que individualiza os organismos para entender como estes funcionam, a ALIFE tenta agrupar sistemas que se comportem como organismos vivos. Esta tentativa de recriar fenômenos biológicos por meios artificiais, segundo Rosenbloom, 1993, irá resultar não somente em uma melhor compreensão do fenômeno da vida como também na aplicação de princípios da biologia na tecnologia de hardware e software, robôs móveis, medicina, fábricas, entre outros.

Em uma primeira aproximação, “Vida” pode ser definida como algo capaz de se reproduzir e de se adaptar a um ambiente e, principalmente, de agir independentemente de agentes externos. Em ALIFE o ambiente é originalmente criado por humanos em um computador. Suas regras são universais e se aplicam mesmo se estiver fora da natureza. A espera por uma ação é justificada por um conjunto de regras estabelecidas pela criação, e, se estas criaturas são capazes de se reproduzir e de sofrer alguma forma de mutação complexa o suficiente para sintetizar evolução ou um padrão de sobrevivência no ambiente encontrado, ela caracterizadamente constitui um protótipo de ALIFE. Para Schalkoff, 1990, o ambiente pode ser definido como um problema computacional tal como a contagem de pessoas em uma plataforma, ou a resolução de um problema matemático, ou pode ser uma área de memória compartilhada com outras criaturas similares.

## 2.1. Inteligência Artificial na linha do tempo

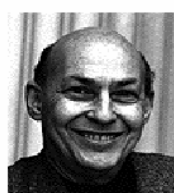
Elementos representativos na conceituação final de Inteligência Artificial e que representaram avanços para a determinação dos procedimentos atuais, são apresentados:

- **1637:** Descartes "Eu penso, portanto eu existo."
- **1726:** Jonathan Swift. A máquina de escrever livros.
- **1811:** Luddite. Automação de controles simples.
- **1818:** "O Frankenstein" de Mary Shelley's.
- **1835:** Relé elétrico foi inventado por Joseph Henry.
- **1847:** Boole desenvolve lógica de símbolos e mais tarde a lógica binária.
- **1855:** Inventada a bomba de mercurio por tubos de vácuo.
- **1859:** Teoria de Darwin "A origem das espécies".
- **1876:** Alexander Graham Bell requer a patente sobre o telefone. Patente 174,465).
- **1879:** Edison inventa a lâmpada elétrica.  
Frege inventa o cálculo predicado.
- **1887:** Primeiro automóvel movido a gás na Alemanha.
- **1901:** "A Interpretação dos sonhos" de Sigmund Freud.
- **1904:** Primeiro tubo de vácuo. Um diodo por John Ambrose Fleming.
- **1913:** Henry Ford sugere a idéia da produção em linha de montagem.
- **1915:** A teoria da relatividade de Einstein.
- **1917:** Karel Capek cria o termo "Robot".
- **1928:** O Teorema Minimax de John von Neuman. Utilizado em programas em grafos de jogos.
- **1930:** Shannon demonstra a lógica de Boole utilizando circuitos elétricos.  
Vannevar Bush controla um Analizador Diferencial Analógico.
- **1931:** Teorema de Gödel.
- **1932:** RCA demonstra o tubo de raios catódicos na tela da TV.



- **1936:** Público de televisão no Reino Unido.  
Church, orientador de Turing inventa o cálculo Lambda, equivalente a máquina apresentada por Turing em sua Tese de Doutorado.
- **1937:** Turing publica "On Computable Numbers".
- **1939:** Dickinson apresenta a memória eletrônica elementar.
- **1940:** Atanasoff e Berry - primeiro computador eletrônico.
- **1940:** Robinson é o primeiro computador operacional baseado em regras, usado para decodificar códigos Nazistas.
- **1940:** Primeira televisão a cores.
- **1941:** Zuse constrói o Z3 na Alemanha, o primeiro computador programável.
- **1943:** Colossus usa tubos eletrônicos para ajudar os Britânicos a quebrar códigos Alemães.  
McCulloch e Pitts propõem a arquitetura de inteligência na rede neural.  
Hebb enuncia sua Lei, base para treinamento em redes neurais.
- **1944:** Aiken completa o 'Mark I', primeiro computador americano programável.
- **1945:** Grace Murray Hopper descobre o primeiro "bug" em 9-SEP-45 15:45.  
Vannevr Bush publica "Como nós pensamos" no "Atlantic Monthly" na forma de hipertexto.
- **1946:** Eckert & Mauchley constroem "ENIAC", o primeiro computador digital programável.  
TV entra na vida Americana.  
John von Neumann publica o artigo EDVAC a respeito de programas armazenáveis.
- **1947:** Fundado o ACM  
Schockley, Brittain e Ardeen inventam o transistor nos laboratórios Bell.
- **1948:** Norbert Wiener's cria o protótipo "Cybernetics".
- **1949:** Wilkes constrói o EDSAC, primeiro programa de computador armazenado.  
Eckert e Mauchley constroem o BINAC de menor porte.  
Orwell publica "1984", no qual descreve que computadores são utilizados para escravizar a população.

- **1949:** Shannon introduz a teoria da Informação.
- **1950:** Isaac Asimov publica "Eu, o Robô".  
Shannon propõe um programa de jogo de xadrez.  
Turing propõe um Teste: "Computing Machinery and Intelligence".  
Um comercial colorido na TV dos Estados Unidos.  
Eckert e Mauchley criam o UNIVAC, primeiro computador comercial.
- **1951:** Census Bureau compra a Remington-Rand UNIVAC.  
EDVAC completado na Universidade da Pennsylvania.  
Fundada a IEEE - Sociedade de Computação.  
A TV dos Estados Unidos envia um sinal transcontinental.
- **1952:** Alan Turing morre.  
CBS usa o UNIVAC para prever a eleição de Eisenhower-Stevenson.  
Primeiro computador usado por DoD (IBM 701).  
Grace Murray Hopper descreve o compilador.  
Introduzido o rádio transistorizado de bolso.  
Rochester constroi o IBM 701.
- **1953:** Watson e Crick descobrem a estrutura química do DNA.
- **1954:** Isaac Asimov, "The Caves of Steel" - Robô de ficção científica.
- **1955:** Newell, Shaw, and Simon desenvolvem IPL11 - Primeira linguagem de IA.  
IBM introduz o primeiro transistor na calculadora 1955: Sperry-Rand merger.
- **1956:** Newell, Shaw, and Simon criam "A Teoria da Lógica" para solucionar problemas matemáticos.
- **1956:** Primeira conferência de McCarthy, Minsky, Newell, and Simon.



Marvin Minsky



Herbert Simon



Allen Newell



John McCarthy

Fig. 2.1. Fotos de Minsky, Simon, Newell e McCarthy

CIA consolida o projeto GAT - máquina de tradução.

Linguagem FORTRAN foi criada pela IBM.

Primeiro sistema comercial bancário – ERMA - Banco da América.

Ulam cria "MANIAC I", programa de xadrez para vencer o ser humano.

- **1957:** Chomsky escreve "Estruturas Sintáticas".  
Newell, Shaw e Simon criaram o Solucionador de Problemas Gerais (GPS), analisador de finais de recursos.  
Fundada a "Digital Equipment Corporation" por Chomsky.
- **1958:** McCarthy introduz teoricamente a linguagem LISP.  
Surge o ALGOL 58.  
Fundada a DARPA.  
Jack St. Clair Kilby inventa o circuito integrado.
- **1959:** Minsky e McCarthy criam o Laboratório de Inteligência Artificial.  
Rosenblatt introduz "Perceptron".  
Programa de Xadrez de Samuel vence o jogo contra os melhores jogadores humanos.  
Kurtz e Kemeny introduzem os sistemas de tempo compartilhado.  
Criada a linguagem COBOL.  
Robert Noyce do TI inventa o circuito integrado independente de Kilby.
- **1960:** Computadores de defesa enganam-se confundindo a lua com mísseis inimigos.  
LINC é o primeiro minicomputador com CRT integral (Lincoln Labs).  
Fitas magnéticas.  
Bar-Hillel publica um artigo descrevendo a dificuldade da tradução pelas máquinas.  
Steve Russel implementa LISP em uma IBM704 como trabalho em curso do MIT de McCarthy.
- **1961:** DEC vende PDP-1.

- **1961:** John Fitzgerald Kennedy propõe o projeto Apollo para o Congresso.
- **1962:** McCarthy vai para Stanford, fundando o Laboratório IA de Stanford em 63.  
Purdue estabelece o primeiro departamento, oferecendo o curso de PhD.  
TECO, o editor de texto de Murphy & Greenblatt, executado em um PDP1 no MIT.  
Primeira indústria de robôs comerciais.  
"Structure of Scientific Revolutions" de Kuhn.  
"Knowledge and Belief" de Hintikka.  
"Possible Worlds Semantics" de Kripke.
- **1963:** ARPA recebe \$2 million cedido para o Laboratorio de AI do MIT.  
Sistema SABRE na American Airlines. Primeiro sistema de reservas aéreas.  
SKETCHPAD de Sutherland: ferramenta de desenho (CAD), forçando a criar o WYSIWYG.  
M. Ross Quillian - Redes semânticas para reconhecimento e representações.  
MIT projeta MAC.  
Testado o "parser" de Susumo Kuno no "Time flies like an arrow".  
"Steps Towards Artificial Intelligence" de Minsky.
- **1964:** DEC PDP-8 é o primeiro computador produzido em massa.  
IBM introduz a série 360.  
Kemeny & Kurtz introduz a linguagem BASIC.  
McLuhan escreve "Understanding Media" que prevê a global vila de inteligência.  
Surgem as linguagens PL/1 e BASIC.  
Estudo de Bobrow resolve problemas de álgebra da escola superior.  
Desenvolvimento do Lisp iniciado em código binário.
- **1965:** Linguagem de programação APL – A Programming Language.  
Buchanan, Feigenbaum e Lederberg iniciam DENDRAL projeto de sistema especialista.

Iva Sutherland demonstra o primeiro visor de cabeça. Realidade virtual.  
 Simon prevê que as máquinas poderão fazer qualquer coisa que o homem possa fazer".

Encontro de Lógica na Universidade Federal Fluminense organizado por Jorge Emanuel Ferreira Barbosa, Castrucci (USP), Doris Aragon, Jorge Muniz Barreto (IME).

Dreyfus sugere novamente a possibilidade da Inteligência Artificial.

- **1966:** Donald Michie funda o Laboratório de Inteligência Artificial de Edinburgo. Sistema especialista ELIZA de Weizenbaum.
- **1967:** Greenblatt's MacHack e Hubert Deyfus projetam Sistemas especialistas. A IBM separa o hardware do software. Papert desenvolve a linguagem LOGO.
- **1968:** HAL estrela no filme 2001 de Clarke and Kubrick. Artigo de Dijkstra CACM "GO TO statement considered harmful". Englebart demonstra o mouse, windows, multiple monitors. Primeiro PhD em Ciência da Computação (Wexelblat at Univ. of Penn.). "Processamento de Informações Semânticas" de Minsky. "The Sound Pattern of English" de Chomsky e Halle. "Philosophical problems from the standpoint of Artificial Intelligence" John McCarthy e Pat Hayes (Situation Calculus).
- **1969:** Tese de Ph.D. de Alan Kay descreve a teoria do computador pessoal. Processo antitruste movido contra a IBM. Planejamento de Green usando um provador de teoremas. A Arte da Programação, Vol I de Knuth. "Perceptrons" de Minsky & Papert (limita uma simples camada de rede neural). UNIX (Thomson e Ritchie pela AT&T). Hearn e Griss definem o standard lisp para portar um sistema de álgebra simbólica reduzida.
- **1970:** Linguagem PROLOG (Colmerauer).

Introduzidos os Floppy disks (disquetes).

Negroponte forma o Grupo de Arquitetura de Máquinas.

Pople e Myers inicia INTERNIST (exame e diagnóstico de doenças humanas).

SHRDLU de Terry Winograd (Processamento de Linguagens Naturais, utilizando blocos).

Arquitetura computacional ARCH de Winston.

- **1971:** Modelo PARRY de Colby.  
Nils Nilsson e Rich Fikes - STRIPS - o primeiro sistema de planejamento.  
ARPA lança o projeto SUR (Spoken Language Understanding) - precursor do HARPY e sistemas HWIM.  
Primeiro microprocessador nos USA (Intel 8088).  
Primeira calculadora de mão: Pocketronic.  
Linguagem Pascal.
- **1972:** Dreyfus publica "What Computers Can't Do".  
Smalltalk desenvolvido pela Xerox PARC (Kay).  
Formado a Cray Research.  
Hewlett Packard introduz a HP-35 por \$395.  
PONG de Nolan Bushell - primeiro vídeo game.
- **1973:** Lighthill report kills é fundada no Reino Unido.  
Schank e Abelson desenvolvem os scripts (linguagens para textos).  
Primeiro monitor bit-mapeado orientado a gráficos.  
Widespread promove a distribuição do UNIX para universidades.  
Xerox PARC constroi "Alto" com o primeiro mouse de mão with first hand-held mouse.
- **1974:** Edward Shortliffe's thesis on MYCIN.  
Criado o primeiro robô controlado por computador.  
"A Framework for Representing Knowledge" de Minsky.  
Estabelecida a rede de trabalho SUMEX-AIM (Aplicação da IA na medicina).

Kowalsky – Queen Mary.

AHL Publica "Creative Computing".

Primeiro SIGGRAPH.

Nelson escreve "Computer Lib".

- **1975:** Cooper & Erlbaum encontrou Nestor para o desenvolvimento de tecnologias de redes neurais.
  - Primeira tiragem da revista BYTE.
  - Primeiro microcomputador - BASIC de Gates e Allen.
  - Primeiro computador pessoal - Altair 8800 (256 bytes de memória).
  - DARPA launches image understanding funding program.
  - Larry Harris funda a Artificial Intelligence Corporation (NLP).
- **1976:** Adventure (Crowther e Woods) - primeiro jogo de adventure.
  - Greenblatt cria a primeira máquina LISP, "CONS".
  - Kurzweil introduz a máquina de leitura.
  - AM de Lenat (Matemática Automatizada).
  - Marr's primal sketch as a visual representation.
  - Super computador Cray-1, 138 megaflops.
  - Artigo Dynabook (Kay e Goldberg).
  - Wozniak e Jobs desenha e controla o Computador Apple Computer.
- **1977:** 3CPO e R2D2 estrela em "Star Wars".
  - Apple II, Radio Shack TRS80, Commodore PET.
  - Conway & Mead codifica desenho VLSI.
  - Primeiro acampamento computadorizado para crianças.
  - Fundada a Microsoft.
- **1978:** Marr e Nishihara propõem esboço 2-1/2 dimensional.
  - Bricklin escreve VisiCalc.
  - Hayes anuncia o Micromodem 100.
  - PROSPECTOR de SRI Lybdenum.
  - Máquinas LISP da Xerox.
- **1979:** Introduzido o "Pac Man".

Compuserve e The Source são fundadas.

ADA.

Pat Hayes: "The Naive Physics Manifesto".

Steve Jobs visita a Xerox PARC.

Raj Reddy funda o Instituto de Robótica em Carnegie Mellon University.

MYCIN é tão bom quanto um médico experiente (Journal of American Medical Association).

- **1980:** Sistemas especialistas com milhares de regras.

Primeira Conferência AAAI de Stanford.

Greenblatt & Jacobson encontra LMI; Noftsker starts Symbolics.

Hofstadter escreve "Gödel, Escher, Bach", vencendo o Pulitzer.

McDermott's XCON para configurar sistemas VAX (DEC e CMU).

Primeira bianual ACM Lisp e Conferência de Programação Funcional.

Protótipo de um conselheiro de "Dipmeter".

Scribe, o primeiro processador de textos.

Xerox, DEC, & Intel anunciam a Ethernet.

- **1981:** Allen Newell "The Knowledge Level".

IBM introduz o computador pessoal (PC).

Kazuhiro Fuchi anuncia o Projeto "Japanese" para a quinta geração.

MITI quer computadores inteligentes por volta de 1990.

Teknowledge fundada por Feigenbaum.

PSL (Portable Standard Lisp Standard e Potavel) roda em uma variedade de plataformas.

Xerox Star (Desktop publishing).

Máquinas Lisp machines da Xerox, LMI, e Simbólicas disponíveis comercialmente, fazendo a tecnologia dinâmica de OOP disponível em uma base difundida.

Definições básicas do Common Lisp é o aspecto comum da família de linguagens - Lisp Machine Lisp, MacLisp, NIL, S-1 Lisp, Spice Lisp, Scheme.



- **1982:** Publicação do Relatório do governo Britânico "Alvey Report" sobre tecnologia de informação avançada on advanced information technology, sugerindo o apoio na IA (Sistemas especialistas) usados na indústria.
  - Formato o ICOT no Japão.
  - John Hopfield ressuscita as redes neurais.
  - PROSPECTOR de SRI encontra um depósito maior de molybdenum.
  - Hayes 300 Smartmodem com conjunto de comandos AT - IBM PC.
  - Kapor desenvolve o "Lotus 1-2-3".
- **1983:** Asimov escreve "Robots of Dawn".
  - Feigenbaum & McCorduck publica "The Fifth Generation".
  - DARPA anuncia a iniciativa de Computadores Estratégicos.
  - intelliGenetics comercializa KEE.
  - Consórcio MCC formado por Bobby Ray Inman.
  - 6.000.000 computadores vendidos no mundo.
  - AT&T breakup.
  - Apple LISA.
  - IBM anuncia o PCjr.
  - Sony anuncia a tecnologia do disco compacto (CD).
- **1984-86:** Corporações investem 50 milhões de dólares em iniciativas sobre a IA.
- **1984:** Publicação do Livro de Steele intitulado "Common Lisp the Language".
  - Chamberlain's RACTER.
  - Doug Lenat inicia o projeto CYC no MCC.
  - Comunidade Européia inicia o programa ESPRIT.
  - GM investe 4 milhões de dólares na Teknowledge.
  - Gold Hill cria o Golden Common LISP.
  - TI vence o contrato do MIT por máquinas LISP sobre símbolos.
  - Apple lança o Macintosh.
  - Chip RAM em megabits da IBM.
  - Lançados os discos Optical.
  - Perez & Rapaport start Neuron Data, selling Nexpert for the Mac.

Phil Cooper funda a Palladian.

- **1985:** GM e Campbell's – Lisp para sistemas especialistas.  
 Kawasaki robot kills Japanese mechanic during malfunction.  
 Fundado o MIT Media Lab.  
 Minsky publica "The Society of Mind".  
 Palladian vende a "Financial Adviser".  
 Teknowledge abandona LISP e PROLOG substituindo por Linguagem C.  
 Xerox vende \$20 milhões em contrato para máquinas LISP.  
 Surge a linguagem C++.  
 Commodore AMIGA, ATARI 520 ST.  
 Interfaces gráficas disponíveis no mercado.  
 Microsoft Windows ships.
- **1986:** X3J13 forms to produce a draft for an ANSI Common Lisp standard.  
 Robô Anderson. Jogador de ping-pong vence jogador humano.  
 Borland oferece Turbo PROLOG for \$99.  
 CMU's HiTech chess machine competes at senior master level.  
 Dallas Police use robot to break into an apartment.  
 Primeira OOPSLA conferencia sobre programação orientada a objetos, na qual CLOS é a primeira referência fora do Lisp.  
 IBM enters AI fray at AAAI, with a LISP, a PROLOG, and an ES shell.  
 Max Headroom.  
 McClelland e Rumelhart's "Parallel Distributed Processing" (Neural Nets).  
 Neural net startup companies appear.  
 OCR uma indústria de cem milh~es de dólares.  
 Grupo PICON-ES abandona a LMI e inicia o projeto Gensym.  
 Paperback Software offers VP Expert for \$99.  
 Teknowledge publica artigos sobre IA.  
 Máquinas "pensantes" introduzidas.  
 Surge o software "Adobe PostScript".  
 Motorola 68020 introduced

- **1987:** Simbologia OODB.
  - Publicado o Lisp Pointers.
  - Sistemas especialistas em linhas de trabalho.
  - AI revende 1.4 bilhões de dólares.
  - NLP revende approximately 80 milhões de dólares.
  - Robotic-vision revende 300 milhões de dólares.
  - DEC's "XCON" configura computadores de trabalho de 300 pessoas using 10.000 regras.
  - Surge o Japan's AFIS (Automated Fingerprint Identification System).
  - LMI files for bankruptcy, other bankruptcies and layoffs follow.
  - "AI Winter"; Lisp-machine. Mercado saturado.
  - Apple introduz "HyperCard".
  - Novos modelos conquistam o mercado.
  - "Pixar" de George Lucas assina contrato com a "Symbolics".
  - Japan desenvolve Identificador de impressões digitais.
- **1988:** Common Lisp development environments on general purpose platforms begin to rival those available on Lisp machines (e.g., native CLOS, preemptive multitasking, full suites of integrated tools, etc.).
  - 386 chip brings PC speeds into competition with LISP machines.
  - Sistemas especialistas arrecadam 400 milhões de dólares.
  - Hillis's "Connection Machine", capacidade de 65.536 cálculos paralelos.
  - Minsky e Papert publicam edição revisada of "Perceptrons".
  - Linguagens orientadas a objetos se firmam.
  - TI anuncia microExplorer (Macintosh with a LISP chip).
  - "Teknowledge" se associa a "American Cimflex".
  - Morris anuncia sua linha de computadores.
  - Fabricação em série de computadores pessoais.
  - Vendidos no ano 4.700.000 microcomputadores, 120.000 minicomputadores e 11.500 sistemas de grande porte.
- **1989:** Coral vendida a Apple. Macintosh Allegro Common Lisp.

Palladian lança sistemas operacionais.

1000 Hospitais americanos agraciados com sistemas especialistas.

Japão lança modelos de tamanhos reduzidos.

- **1990:** Steele publica a segunda edição de "Common Lisp: the Language".  
AI Corp vai a público.  
Symbolics Lisp Users Group (Grupo SLUG) vota para expandir a Associação de usuários de Lisp na conferência anual.  
ESPRIT. Simulador de pesquisas e orçamentos.  
Fundação MacArthur premia Richard Stallman 240 mil dólares pelo espírito criador.  
Novos PC's, NeXT's, Mac's SUN's, DEC's.
- **1991:** KnowledgeWare cancela a oferta para comprar a IntelliCorp.  
Formado o Conselho Nacional de Redes Neurais por ocasião do Workshop de redes neurais na PUC/RJ.
- **1992:** Apple Computer introduz o Dylan, uma linguagem na família LISP.  
X3J13 cria um documento propondo uma padronização Norte americana para o Common Lisp.  
Projeto da quinta geração Japonesa é finalizado.  
Inicia o Projeto Japonês "Real World Computing Project".  
Mais de 1.000 variedades de vírus de computadores.
- **1993:** Kurzweil AI vai a público.  
Criado o IBM/Apple PowerPC.
- **1994:** Franz Inc. anuncia o AllegroStore OODB.  
Real-time CLOS de Harlequin é usado no anúncio da AT&T para sistemas.  
Máquinas "pensantes".  
Projetado ANSI Common Lisp becomes the first ANSI-standard OOPL.  
Departamento de Justiça Norte Americano move ação anti-trust.  
III Congresso Brasileiro de Redes Neurais em Santa Catarina, promovido por J. M Barreto e L. Caloba.

## 2.2. Atividades Psíquicas

Uma vez que as definições de IA envolvem termos como Faculdades Mentais, tarefas desempenhadas por pessoas, coisas que tornam as pessoas inteligentes e atividades mentais humanas em geral, é interessante que se considerem, inicialmente, os aspectos psicológicos da inteligência humana, ainda que o objetivo final não seja reproduzi-los. Segundo os preceitos da Psicologia (Paim, 1986) e da psiquiatria (Ey, Bernard e Brisset, 1985), as atividades psíquicas (do grego, Psykhe significa alma) fundamentais são:

**Sensação:** Fenômeno elementar resultante de estímulos mecânicos, físicos, químicos ou elétricos sobre o organismo. As sensações podem ser externas (visuais, auditivas, gustativas, olfativas ou táteis) ou internas (proprioceptivas equivalem a forças e posições das partes do corpo, cinestésicas equivalem a movimento das partes do corpo, ou autônomas equivalem a funções dos órgãos internos do corpo).

**Percepção:** É a tomada de conhecimento de um objeto exterior considerado real. As sensações são internas mas as demais condições são conhecidas.

**Representação:** É a organização das imagens de objetos. Corresponde ao processo de modelamento em engenharia. As imagens visuais poderiam ser comparadas a fotografias que representam, no papel, algumas características (em termos morfológicos) do objeto fotografado.

**Conceituação:** Em latim, Concepto significa reunido. Construção simbólica que busca captar a essência dos objetos, agrupando-os em classes. É uma das operações intelectuais básicas: Conceituação, juízo e raciocínio.

**Juízo:** É a capacidade de exprimir os vínculos e as relações entre os fatos e os objetos da natureza. Corresponderia à capacidade de qualificar os fenômenos e os objetos do sistema, segundo as leis físicas específicas.

**Raciocínio:** É a concatenação disciplinada dos juízos, de modo a gerar novos juízos. A capacidade de dedução é a grande característica.

**Memória:** É a capacidade de armazenamento de informações para utilização posterior. Uma alteração do estado da memória é a amnésia, que é a incapacidade de acessar os dados armazenados. Em contraposição a amnésia existe o fenômeno do tipo “Dejà vu”, onde a pessoa relata lembrar-se de algo não antes visto.

**Atenção:** É a capacidade de concentrar a capacidade psíquica sobre um dado estímulo que a solicita. Um sistema de visão ativa por computador pode concentrar a sua atenção sobre um determinado alvo a ser atacado.

**Consciência:** É um complexo de fenômenos psíquicos que permite conhecer a si próprio e ao mundo. Configuram alteração do estado de consciência o coma ou rebaixamento profundo da consciência, a possessão e o transitivismo. É questionável se uma máquina poderia manifestar genuinamente tal atividade psíquica, embora possa, eventualmente, produzir respostas compatíveis com a sua identidade através de programas que consultam base de dados.

**Orientação:** É a capacidade de o indivíduo ter consciência de sua situação temporal e espacial. Uma alteração do estado de orientação é a desorientação amnésica, ou seja, a incapacidade de assimilar as relações entre os acontecimentos em termos de seqüência e instantes de ocorrência ou do local de ocorrência.

**Afetividade:** É a capacidade de experimentar sentimentos ou emoções, correspondendo à valorização qualitativa de determinados estados do indivíduo. São exemplos de alterações do estado de afetividade a apatia, a angústia e os tiques.

**Volição:** É o elemento psíquico que leva a iniciar e realizar atividades voluntárias. Alterações no estado de volição características são o estupor e os tiques.

**Linguagem:** É o mecanismo que permite a expressão simbólica, tipicamente verbais ou gráficas, onde as alterações mais comuns na linguagem são a dislalia, a afasia e a logorréia.

## **2.3. Ferramentas utilizadas em IA**

Em princípio, o estudo da Inteligência Artificial poderia incluir considerações que abrangem desde os estudos dos filósofos da Grécia antiga, como Aristóteles, devido a sua contribuição à Lógica, até os modernos pensadores deste século. Destacam-se as ferramentas mais empregadas atualmente:

### **2.3.1. Sistemas de produção**

São os sistemas que utilizam regras, usualmente do tipo “Se – Então” aliado a uma base de dados de conhecimentos e mecanismos de controle, indicando a sequência de regras a serem casadas com a base de dados e, na existência de conflitos, providenciam sua resolução. Caracterizam desta maneira a heurística e os conhecimentos de especialistas humanos, através da aplicação de conceitos de lógica.

Os cálculos de predicados são intimamente ligados aos sistemas de produção uma vez que se caracterizam como sub-área da Lógica.

O desenvolvimento da Lógica se deve a grandes pioneiros como G. Frege (1848-1925), que foi o primeiro a distinguir claramente as premissas e as regras de inferência. Outro cientista que muito contribuiu foi A. Tarski com sua metodologia, em 1941, bem como R. Carnap, em 1950, com a sintaxe e K. Gödel, 1930, com a metalógica.

### **2.3.2. Lógica nebulosa**

Aristóteles, filósofo grego (384 - 322 a.C.), foi o fundador da ciência da lógica e estabeleceu um conjunto de regras rígidas de modo tal que conclusões pudessem ser aceitas e logicamente válidas. O emprego da lógica de Aristóteles levava

a uma linha de raciocínio lógico baseado em premissas e conclusões. Como um exemplo: se é observado que "todo ser vivo é mortal" (premissa 1), a seguir é constatado que "Sarah é um ser vivo" (premissa 2), como conclusão temos que "Sarah é mortal".

Desde então, a lógica Ocidental, assim chamada, tem sido binária, isto é, uma declaração é falsa ou verdadeira, não podendo ser, ao mesmo tempo, parcialmente verdadeira e parcialmente falsa. Esta suposição e a lei da não contradição, cobrem todas as possibilidades, formando a base do pensamento lógico Ocidental.

A Lógica Nebulosa, ou Fuzzy, viola estas suposições. Um sim ou um não como resposta a estas questões é, na maioria das vezes, incompleta. Na verdade, "entre a certeza de ser e a certeza de não ser, existem infinitos graus de incerteza" (Sandri, 1996). Esta imperfeição intrínseca à informação representada numa linguagem natural foi tratada matematicamente com o uso da teoria das probabilidades. Contudo, a Lógica Nebulosa, com base na teoria dos conjuntos nebulosos (Fuzzy Set), tem se mostrado mais adequada para tratar imperfeições da informação do que a teoria das probabilidades.

A Lógica Nebulosa encontra-se entre as técnicas mais recentes de Inteligência Artificial, também conhecida como Conjuntos Difusos. Este termo, a princípio, nos convida a pensar em algo confuso (nebuloso), porém, atualmente, ele é bastante direto. Essa técnica, muito usada no Japão, é fruto da tão esperada quinta geração dos computadores.

A Lógica Nebulosa consiste em aproximar a decisão computacional da decisão humana, tornando as máquinas mais capacitadas a executar seu trabalho. Isto é feito de forma que a decisão de uma máquina não se resume apenas a um "sim" ou um "não", mas também tenha decisões "abstratas", do tipo "um pouco mais", "talvez sim", e outras tantas variáveis que representem possíveis decisões humanas. Ela se constitui em um modo de interligar inerentemente processos analógicos que se deslocam através de uma faixa contínua para um computador digital que pode ver coisas com valores numéricos bem definidos (valores discretos).

Uma das principais potencialidades da Lógica Nebulosa, quando comparada com outros esquemas que tratam com dados imprecisos como redes neurais, é que



suas bases de conhecimento, as quais estão no formato de regras de produção, são fáceis de examinar e entender (Pedrycz, 1993). Este formato de regra também torna fácil a manutenção e a atualização da base de conhecimento.

O conceito de Conjunto Nebuloso foi introduzido, em 1965, por Lotfi A. Zadeh (Universidade da Califórnia, Berkeley). A ele é atribuído o reconhecimento como grande colaborador do Controle Moderno. Em meados da década de 60, Zadeh observou que os recursos tecnológicos disponíveis eram incapazes de automatizar as atividades relacionadas a problemas de natureza industrial, biológica ou química, que compreendessem situações ambíguas, não passíveis de processamento através da lógica computacional fundamentada na lógica booleana. Procurando solucionar esses problemas o Prof. Zadeh publicou, em 1965, um artigo resumindo os conceitos dos Conjuntos Nebulosos, revolucionando o assunto com a criação de Sistemas Fuzzy, King (1977).

Em 1974, o Prof. Mamdani, do Queen Mary College, Universidade de Londres, após inúmeras tentativas frustradas em controlar uma máquina a vapor com tipos distintos de controladores, incluindo o PID, somente conseguiu fazê-lo através da aplicação do raciocínio Nebuloso.

Esse sucesso serviu de alavanca para muitas outras aplicações, como em 1980, no controle Nebuloso de operação de um forno de cimento. Vieram em seguida, várias outras aplicações, destacando-se, por exemplo, os controladores Nebulosos de plantas nucleares, refinarias, processos biológicos e químicos, trocador de calor, máquina diesel, tratamento de água e sistema de operação automática de trens. No Brasil, o primeiro trabalho de que se tem conhecimento foi a Dissertação de Mestrado de Ricardo Tanscheit na PUC/RJ (Tanscheit, 1978).

O desenvolvimento de técnicas de Inteligência Artificial (IA) nos últimos anos, ocupa cada vez mais posição de destaque em pesquisas na área de controle de processos industriais e, aos poucos, começam a ser implantadas em Instalações industriais com enorme sucesso. Dentre as técnicas mais utilizadas, além do Controle Nebuloso, pode-se destacar as Redes Neurais aplicadas a sistemas de controle, que estão atualmente em tamanha evidência que os japoneses as consideram como duas das mais promissoras técnicas para o século XXI.

A primeira aplicação da Lógica Nebulosa bem sucedida foi no desenvolvimento de controladores industriais. Controladores que se baseiam na Lógica Nebulosa são chamados de Controladores Nebulosos. Não é preciso conhecer muito matemática ou em profundidade a teoria de controle para se desenvolver uma aplicação em controle. Controladores nebulosos tratam igualmente sistemas lineares e não lineares, além de não requererem o modelamento matemático do processo a ser controlado. Isto tem sido, sem dúvida, o grande atrativo dos Controladores Nebulosos.

Sistemas baseados na Lógica Nebulosa têm mostrado grande utilidade em uma variedade de operações de controle industrial e em tarefas de reconhecimento de padrões que se estendem desde reconhecimento de texto manuscrito, até a avaliação de crédito financeiro. Existe também um interesse crescente em se utilizar Lógica Nebulosa em sistemas especialistas para torná-los mais flexíveis. No Japão, a Lógica Nebulosa já se faz presente no dia a dia do setor industrial e muitos produtos comerciais já se encontram disponíveis.

## **2.4. Aplicações típicas em IA**

### **2.4.1. Jogos**

Em 1880 L. Torres y Quevedo construiu um dispositivo que podia vencer finais de jogos de xadrez jogando contra um adversário humano. Em 1951, Alan Turing programou a sua máquina desenvolvida em 1946, para jogar xadrez. Essa máquina, chamada de *automatic computing engine*, havia sido inicialmente concebida para decodificação de mensagens. Com esse sistema, Turing jogou contra Alick Glennie, tendo abandonado após a perda da rainha. Em 1949, Claude E. Shannon descreveu formas de programar computadores para jogar xadrez, já incorporando discussões sobre o balanceamento entre força bruta (busca exaustiva) e a heurística. Em 1956, Ulam e Stein programaram uma versão simplificada de xadrez, com tabuleiro de 6 X 6, sem bispos, sem roque e com peões limitados a avançar apenas uma casa no primeiro

movimento. Esse programa, MANIAC I, foi melhorado e se tornou o primeiro a derrotar um ser humano.

Em 1958, um programa desenvolvido por T. Arbuckle e M. A. Belsky em um computador IBM 704, atingiu níveis compatíveis com jogadores amadores de xadrez. Em 1965, o programa MAC HACK, no PDP-6, já era capaz de vencer 80% dos jogadores não participantes de torneios. A versão MAC HACK VI, em 1968, possuía rating de 1.500 pontos.

Em 1970, o primeiro campeonato de xadrez por computador foi realizado nos Estados Unidos, sendo vencido pelo CHESS 3.0 de David Slate, Larry Atkin e Keith Gorlen. O primeiro campeonato mundial de xadrez por computador ocorreu em Estocolmo, 1974, vencido pelo programa KAISSA (Deusa do xadrez), no qual havia contribuições do ex-campeão mundial Mikhail Botvinnik. O segundo campeonato mundial de xadrez por computador, realizado em Toronto, foi vencido por CHESS 4.6.

Em 1978, o Mestre Internacional David Levy ofereceu 1.250 libras para o programa que pudesse vencê-lo. Levy jogou contra CHESS 4.7. Foi a primeira vez que um programa de computador venceu um Mestre Internacional. Posteriormente Levy derrotou o CRAY BLITZ, vencedor do Campeonato Mundial de Xadrez por Computador, de 1983.

Em 1985 surgiu o HITECH, projetado por Hans Berliner, Carl Ebeling e Murray Campbell, utilizando um processador para cada quadrado do tabuleiro. Mesmo com os fantásticos progressos, DEEP THOUGHT, desenvolvido por Feng Hsiung Hsu, Thomas Anantharaman e Andreas Nowatzyk, foi derrotado pelo campeão mundial Garry Kasparov, em 1989.

Em 1996, Kasparov enfrentou o DEEP BLUE da IBM, que ganhou a primeira partida, empatou duas partidas e perdeu três (portanto, Kasparov venceu o torneio na pontuação total). Em 1997, Kasparov foi derrotado em um novo torneio por uma versão mais avançada do DEEP BLUE.

### 2.4.1.1. Grafos de jogos

Os jogos foram a primeira atividade que recebeu atenção dos pesquisadores em IA, principalmente por sua simplicidade de programação, onde as regras são facilmente formalizáveis e pela facilidade de determinar a qualidade do programa, em comparação com outros programas ou seres humanos. “Existem basicamente duas representações de grafos num computador e correspondem às representações seqüenciais ligadas a listas lineares” (Terada, 1991).

Utilizando o procedimento “Minimax” nem sempre é possível ou conveniente percorrer todo o grafo de jogo para se obter uma estratégia. Nesse caso, nos contentamos em obter, dada uma posição, um “bom” lance para essa posição. O procedimento considera basicamente dois jogadores, Min e Max, e constrói a árvore de busca, completa, até uma profundidade  $P$ ; utiliza, então, uma função de avaliação  $f$ , nos vértices terminais, definida de forma que Max prefira os vértices com valores máximos de  $f$ , enquanto Min dá preferência aos valores mínimos de  $f$ ; esses valores são propagados, então, até a raiz da árvore, na seguinte forma:

- Se  $V$  é um vértice Max,  $f(V) = \max \{f(\text{filhos de } V)\}$
- Se  $V$  é um vértice Min,  $f(V) = \min \{f(\text{filhos de } V)\}$

### 2.4.1.2. O jogo da velha

Função de avaliação:	$f(V)$	$-\infty$ , se $V$ representa uma situação de vitória para Min
		$\infty$ , se $V$ representa uma situação de vitória para Max
		$n^\circ$ de fileiras abertas para Max - $n^\circ$ de fileiras abertas para Min

Se Max utiliza o símbolo X e  
Min utiliza o símbolo 0, temos,  
na posição  $V$ :

$V =$	0		
	X		

6 fileiras abertas para Max e 4 fileiras abertas para Min; portanto,  $f(V) = 6 - 4 = 2$ .  
Fica assim definido o grafo para o Jogo da Velha até o nível 2, (Pinho, 1999).

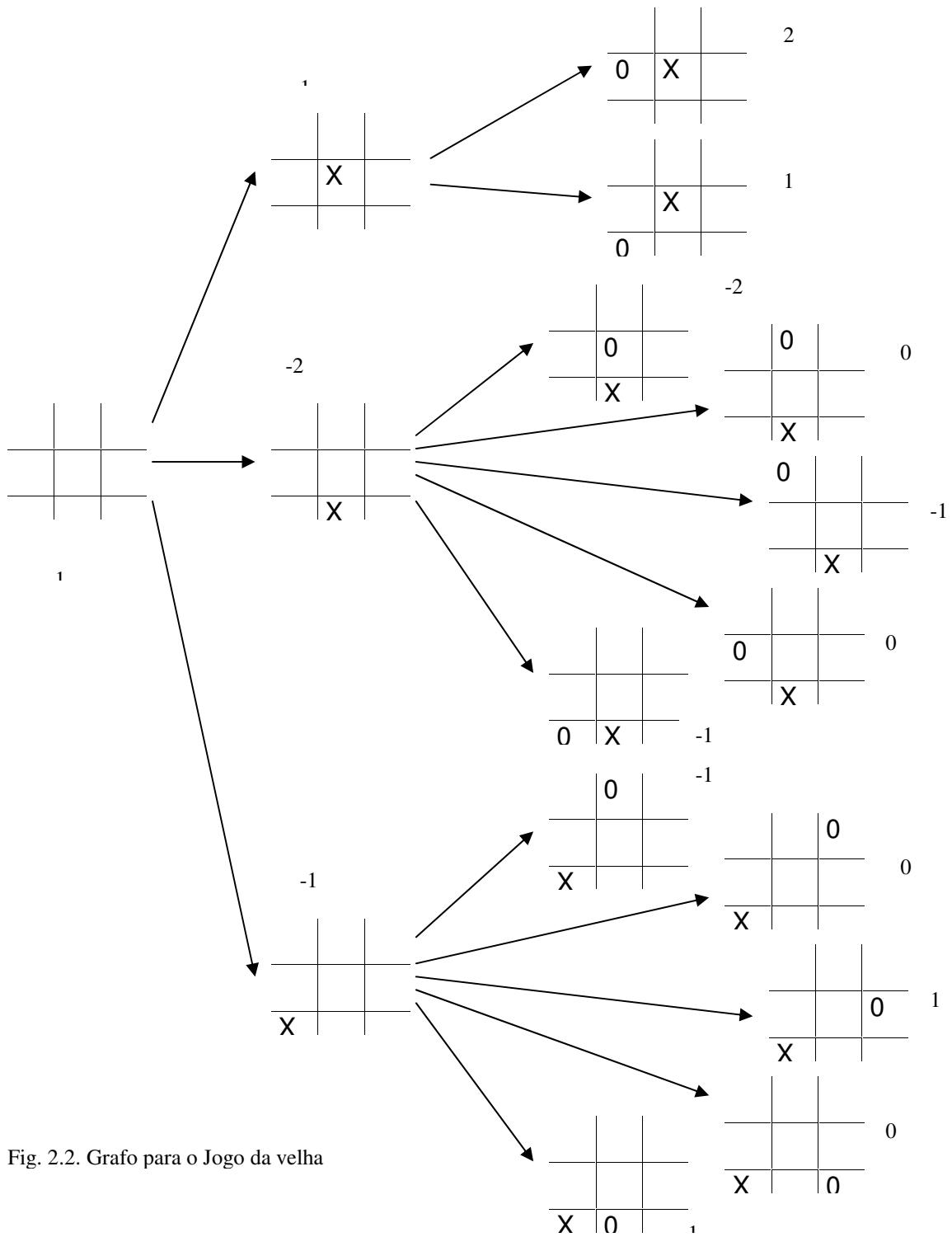


Fig. 2.2. Grafo para o Jogo da velha

## **2.4.2. Prova automática de teoremas**

Esta área de aplicação da IA está intimamente ligada ao desenvolvimento da Teoria de Lógica, uma vez que teoremas são, enfim, sentenças lógicas. Portanto, contribuições como a de J. Herbrand, 1930, em relação a algoritmos para prova e de J. A. Robinson, 1965, que propôs um modo eficiente de inferência automática usando resolução, foram decisivas. A heurística também desempenha um papel importante, como proposto por A. Newel, em 1963, no provador de teorema Logic Theorist. Textos de pesquisadores como N. J. Nilsson, 1971, R. Kowalski, 1979 e D. Loveland, 1978, apresentam bases sólidas para aplicação em provas de teoremas.

Outras Lógicas aplicadas, umas com muita e outras com pouca difusão, são temáticas importantes quando tratamos, matematicamente, das funções de aprendizado. Algumas lógicas pertinentes são destacadas: modal, temporal, combinatória, condicional, construtiva, cumulativas, deontic, dinâmicas, epistêmicas, erotetic, livre, infinitária, intencional, intuicionista, linear, de muitas classes, de muitos valores, não monotônicas, parcial, da quantidade, relevante, severa, paraconsistentes, entre outras.

## **2.4.3. Sistemas especialistas**

Sistemas especialistas, SE's, são concebidos para atuarem como consultores altamente qualificados, em uma determinada área do conhecimento. Devem, portanto, fornecer diagnósticos, condutas, sugestões ou outras informações úteis, obtidas a partir de dados fornecidos localmente e de conhecimentos armazenados previamente. Um sistema especialista pode ser utilizado para auxiliar médicos, operadores de máquinas ou clientes de uma loja. Pode também ser empregado em sistemas de ensino assistido por computador. Enfim, sistemas especialistas permitem que conhecimentos possam ser disseminados e utilizados por um grande número de pessoas, sem a necessidade da presença do especialista. Permite ainda, que conhecimentos de vários especialistas possam ser amalgamados em uma só base de dados (BD).

São descritos a seguir, alguns dos sistemas pioneiros e mais conhecidos:

## MYCIN

Sistema especialista desenvolvido pela Universidade de Stanford na década de 70, pela equipe liderada por B. G. Buchanan e E H. Shortliffe, é considerado o primeiro sistema especialista a alcançar de fato, um padrão de desempenho compatível com o ser humano. Aplicado ao campo da medicina, identificava agentes infecciosos, sugerindo condutas terapêuticas.

O Mycin foi desenvolvido, parcialmente, com o objetivo de simular como os especialistas humanos faziam seus diagnósticos baseados em informações parciais. Contudo, diante de um problema potencialmente importante em termos práticos, alguns médicos fazem diagnósticos sem uma ferramenta disponível para efetivar um tratamento mais adequado. A grande justificativa deste impedimento se apoia em questões éticas e legais impostas à utilização de computadores na Medicina. Quem poderia se responsabilizar em casos de diagnósticos incorretos?

De qualquer maneira o Mycin apresenta seu conjunto de regras na forma de IF-THEN, conforme o procedimento:

*IF the infection is primary-bacteremia  
AND the site of the culture is one of the sterile sites  
AND the suspected portal of entry is the gastrointestinal tract  
THEN there is suggestive evidence (0.7) that infection is bacteroid.*

O nível 0.7 no comando acima representa o grau de certeza para a conclusão atingida, onde a evidência de certezas e de incertezas determinadas pelos bits, combinados com as regras implementadas para a conclusão.

Escrito em LISP e com suas regras representadas por expressões LISP, suas ações poderiam concluir sobre problemas ou simplesmente apresentar arbitrariamente expressões LISP como resposta. Isso permite grande flexibilidade, mas eliminaria a modularidade e a clareza de um sistema baseado em regras.

O Mycin, pelo seu pioneirismo como Sistema Especialista utilizado em pesquisas, trouxe consigo alguns problemas que só foram solucionados mais tarde através de sua utilização em arquiteturas mais sofisticadas. Um desses problemas clássicos foi a pequena confusão de regras de domínio ou condições iniciais, como por exemplo, não checava se o paciente era criança ou adulto para o diagnóstico de problemas de alcoolismo, entre outros.

Outras versões do Mycin surgiram posteriormente. O Neomycin, o Emycin e o Puff, sistemas especialistas desenvolvidos para diagnóstico e tratamento de problemas cardíacos.

## **DENDRAL**

Sistema especialista no campo da análise química que permite analisar a estrutura molecular de substâncias a partir de dados de espectroscopia, desenvolvido por J. Lederberg, E. Feigenbaum e B. Buchanan, em 1965, é programado em LISP e gera, utilizando heurísticas sobre o histograma de intensidade versus números de massa, as estruturas candidatas para a substância que está sendo analisada. Posteriormente, o programa simula a saída do espectrômetro de massa para as estruturas candidatas e produz como saída aquela que se casar melhor com os dados empíricos.

## **MACSYMA**

Concebido por C. Engleman, W. Martin e J. Moses no MIT, em 1968, tem recebido desenvolvimento contínuo. É um sistema codificado em LISP e permite processar simbolicamente problemas matemáticos grandes e complexos.

## **HEARSAY**

Nas versões I e II, foram desenvolvidos pela Universidade de Carnegie-Mellon, iniciado nos fins dos anos 60 e concluído nos anos 70, concebido como parte de um sistema para reconhecimento de linguagem natural falada. Programado em SAIL, HEARSAY recebe como entrada sinais digitalizados de voz e gera como saída



hipóteses do que teria sido falado. Utiliza conceitos como o de Quadro Negro e, posteriormente, algumas idéias foram utilizadas para o HEARSAY III.

### **INTERNIST ou CADUCEUS**

Sistema Especialista demonstrado em 1974, foi desenvolvido por H. E. Pople Jr. E J. D. Myers da Universidade de Pittsburgh. Programado em INTERLISP, é um sistema especialista no domínio da Medicina Interna. Recebe os dados constantes da anamnese e do exame físico e fornece uma hipótese diagnóstica.

### **PROSPECTOR**

É um sistema especialista no campo da geologia, em investigação de possíveis depósitos de minérios. Desenvolvido na década de 1970 por uma equipe do Instituto de Pesquisas de Stanford, M. Einandi. Também programado em LISP, utiliza uma base de conhecimentos empregando redes semânticas. As inferências são organizadas por regras de produção.

### **PUFF**

Sistema especialista no campo da saúde constituído em 1979, por J. S. Aikins, J. C. Kunz, E. H. Shortliffe e R. J. Fallat de Stanford, com base no EMYCIN. Permite a interpretação de medidas e testes respiratórios, fornecendo indicativos sobre funções pulmonares. A Segunda versão do PUFF foi escrita em BASIC para o minicomputador PDP-11.

## **2.4.4. Compreensão de linguagem natural**

As linguagens são conjuntos de seqüências de símbolos ou caracteres, organizadas de acordo com regras definidas a priori. A linguagem formal, LF é utilizada para programação de computadores ou para construção de fórmulas no estudo da lógica. A Linguagem Natural, LN, ou informal é utilizada corriqueiramente por humanos.

As regras para construção das sequências de caracteres formam a gramática, como segue:

<sentença>	= <sujeito><verbo><objeto>
<sujeito>	= <artigo><substantivo>
<verbo>	= COMPROU
<objeto>	= <artigo><substantivo>
<artigo>	= O
<artigo>	= UM
<substantivo>	= PROFESSOR
<substantivo>	= LIVRO

Uma sequência válida gerada por elas: O PROFESSOR COMPROU UM LIVRO. É interessante observar que essas mesmas regras permitem gerar: O LIVRO COMPROU UM PROFESSOR. Sintaticamente, ambas as frases são corretas.

Entidades como <sujeito> ou <artigo> são chamadas de símbolos não terminais e LIVRO ou COMPROU são chamadas de símbolos terminais. As regras da gramática são chamadas de produções. A entidade <sentença> é, intuitivamente, um não terminal “mais nobre”, uma vez que as demais são partes dela. No caso <sentença> recebe o nome especial de símbolo inicial.

Nessas condições, uma classe de gramática muito importante, pode ser definida: Gramática livre de contexto. É aquela especificada por:

- i) um conjunto finito de não terminais;
- ii) por um conjunto finito de terminais que é disjunto do conjunto de não terminais;
- iii) um conjunto finito de produções na forma  $\langle A \rangle = \alpha$ , onde  $\langle A \rangle$  é um não terminal e  $\alpha$  é uma sequência de terminais e não terminais;
- iv) o símbolo inicial é um não terminal.

Essa classe é geral o suficiente para gerar a título de ilustração as expressões aritméticas e as expressões em LISP. No caso particular de uma gramática, onde as produções possuem no lado direito apenas terminais ou um terminal seguido de um

único não terminal, diz-se tratar de gramática regular. Para elas, há mecanizações computacionais eficientes (empregando autômata) para o seu reconhecimento e, em decorrência, são amplamente utilizadas em computação.

Por outro lado, as linguagens naturais exibem dificuldades de natureza sintática, semântica e pragmática, como observadas por Chomsky, em 1969, e Saussure, em 1988. Apresentar evidências para essa dificuldade é tarefa fácil, como mostram as sentenças:

1. “Pão com manteiga é melhor que nada.  
Nada é melhor que férias.  
Pão com manteiga é melhor que férias.” **(transitividade)**
2. “Quanto dinheiro você tem no bolso?  
Não tenho nada.” **(dupla negação)**
3. “Você tem uma moeda? Sim. **(embora tenha 10)**
4. “Você tem um dedo? Não.” **(embora tenha 10)**
5. “Aquele filme é um abacaxi e os autores só falam abóbora...”
6. “Estou nadando em dinheiro.” **(será que dinheiro é líquido?)”**

Utilizando técnicas diversas, foram desenvolvidos programas de computador que aparentam compreender alguma linguagem natural. Alguns desses programas são:

- ELIZA de Weizenbaum em 1966 (terapista Rogeriano);
- PARRY escrito por Colby em 1975 (modelo de personalidade paranóide);
- STUDENT de Bobrow em 1968 (problemas com palavras);
- O texto de Schank e Abelson em 1977 (busca conectar estratégias de solução de problemas como compreensão de linguagem);
- Trabalhos pioneiros na geração de histórias foram realizadas por Klein em 1976.

Freqüentemente são apresentados muitos problemas ao se tratar das linguagens naturais. Os principais problemas, segundo Barreto, 2000, são:

- Separação de uma frase falada em palavras.
- Identificação das palavras na linguagem falada.
- Identificação dos símbolos na linguagem escrita.
- O problema da determinação da estrutura da frase.
- Determinação do contexto da frase em um discurso.
- A representação do modelo conceitual do significado de um discurso.
- A manipulação do conceito de um discurso, integrando-o a conceitos já armazenados.
- A geração de um novo discurso e de sua representação interna.
- A produção de um novo discurso.

### **2.4.5. Representação de incertezas**

A imperfeição da informação é geralmente conhecida na literatura de sistemas baseados em conhecimento como *incerteza*. No entanto, este termo é muito restritivo; o que se convencionou chamar tratamento de incerteza pode, na verdade, estar endereçando outras imperfeições da informação, como imprecisão, conflito, ignorância parcial, etc.

Suponhamos, por exemplo, que queiramos descobrir a que horas começa um determinado filme. Algumas das respostas que podemos obter são:

- Informação perfeita: O filme começa às 8h 15min.
- Informação imprecisa: O filme começa entre 8h e 9h.
- Informação incerta: Eu acho que o filme começa às 8h (mas não tenho certeza).
- Informação vaga: O filme começa lá pelas 8h.
- Informação probabilista: É provável que o filme comece às 8h.
- Informação possibilista: É possível que o filme comece às 8h.
- Informação inconsistente: Maria disse que o filme começa às 8h, mas João disse que ele começa às 10h.

- Informação incompleta: Eu não sei a que horas começa o filme, mas usualmente os filmes neste cinema começam às 8h.
- Ignorância total: Eu não faço a menor idéia do horário do filme.

As informações que podemos obter podem, portanto, variar de perfeitas, quando descobrimos exatamente o que queremos saber, a completamente imperfeitas, seja pela total ausência de informações ou por informações conflitantes. O mais interessante aqui é que, mesmo lidando diariamente com o tipo de informações acima, conseguimos tomar decisões razoáveis. Para tanto, nós, de alguma forma, encontramos um modelo adequado para representar a informação que obtivemos e a tratamos segundo o modelo escolhido. O mesmo deve, ou pelo menos deveria, ocorrer com sistemas baseados em conhecimento, em face de informações imperfeitas.

A informação incerta pode ser tratada tanto pelas teorias de probabilidades, possibilidades ou evidência, ou por modelos "ad hoc". As informações inconsistentes e aquelas que chamamos, um pouco impropriamente, incompletas, podem ser tratadas por lógicas não clássicas como a paraconsistente e a lógica de quatro valores no primeiro caso, e as lógicas não monotônicas no segundo caso, como a lógica de "default" e a circunscrição.

São descritos a seguir os modelos numéricos mais conhecidos para a representação da informação imperfeita: os modelos nebuloso, possibilista e da evidência.

## **Modelo nebuloso**

*A teoria dos conjuntos nebulosos* é o modelo mais tradicional para o tratamento da informação imprecisa e vaga. Este modelo, tem por objetivo permitir graduações na pertinência de um elemento a uma dada classe, ou seja de possibilitar a um elemento pertencer com maior ou menor intensidade àquela classe. Basicamente, isso se faz quando o grau de pertinência de um elemento do conjunto, que na teoria "clássica" dos

conjuntos assume apenas os valores 0 ou 1, passa a ser dado por um valor no intervalo dos números reais  $[0,1]$ .

Dado um universo de discurso  $X$ , um subconjunto nebuloso  $A$  de  $X$  é definido por uma função de pertinência que associa a cada elemento  $x$  de  $X$  o grau  $\mu_A(x)$ , compreendido entre 0 e 1, com o qual  $x$  pertence a  $A$ .

Na teoria dos conjuntos nebulosos, a negação é implementada por uma família de operadores, sendo que o mais comumente utilizado é dado por  $\mu_{\neg A}(x) = 1 - \mu_A(x)$ . Ou seja, se  $\mu_A(x)$  representa a negação de  $A$  no universo  $Y$ , então utilizando a negação  $\mu_{\neg A}$  temos  $\mu_{\neg A}(x) = 1 - \mu_A(x)$ .

Nesta teoria, a intersecção é implementada por uma família de operações - chamadas *T-normas* - e a união é implementada por outra família de operações - chamadas *T-conormas*.

O conjunto das T-normas e T-conormas formam as *normas triangulares*. Cada norma triangular verifica, para todos os  $x, y, z$  em  $[0, 1]$ , propriedades abaixo:

Comutatividade:  $\mu(x,y) = \mu(y,x)$

Associatividade:  $\mu(x,(\mu(y,z))) = \mu(\mu(x,y),z)$

Monotonicidade:  $x \leq z$  e  $y \leq t$  então:  $\mu(x,y) \leq \mu(z,t)$

Os *operadores de implicação* são usados para modelar regras de inferência do tipo "Se premissa então conclusão". Seja o grau de compatibilidade entre as condições estabelecidas na premissa e os valores encontrados na realidade e  $C$ , definido em  $Z$ , o conjunto nebuloso presente na conclusão da regra. Então, para verificarmos o grau com que a premissa implica a conclusão, dados os valores encontrados na realidade, verificaremos o quanto implica  $\mu_C$ , verificando para todo.

As regras de controle nebuloso são usualmente do tipo:

$$R_j : \text{Se } x_1 \text{ é } A_{ij} \text{ e } x_n \text{ é } A_{nj} \text{ então } y \text{ é } B_j.$$

onde os  $A_{ij}$  e  $B_j$  são conjuntos nebulosos, e a implicação é implementada como uma função de implicação nebulosa. Em cada ciclo do processo, os valores das variáveis  $x_{ij}$  são medidos e então comparados aos conjuntos nebulosos  $A_{ij}$  nas regras, gerando

uma medida da adequação dos valores medidos à premissa da regra (utilizando uma T-norma para implementar o conectivo “e” na premissa das regras).

## **Modelo possibilista**

A *teoria de possibilidades* é um modelo que alia um grande poder de expressividade com uma grande flexibilidade para o tratamento da informação incerta. Neste modelo, a informação fornecida por uma fonte de conhecimento sobre o verdadeiro valor de uma variável  $x$  em um universo de discurso  $X$  é codificado sob a forma de uma distribuição de possibilidades. Para qualquer valor de  $x$ , reflete até que ponto é possível que  $x$  seja igual a  $x_i$  (partindo do pressuposto de que se tem somente um valor verdadeiro).

Estes valores são supostamente mutuamente exclusivos, pois  $x$  toma somente um valor (seu valor verdadeiro), que pertence a um conjunto universo  $X$  dado. Normalmente, assume-se também que existe ao menos um valor considerado como completamente possível de ser o verdadeiro valor de  $x$ . Isto se traduz pela condição de normalização:  $\sum_{x \in X} p(x) = 1$ . É importante notar que nada impede que valores distintos de  $X$  possam ser considerados completamente possíveis, simultaneamente.

Conhecendo-se a distribuição de possibilidades, a verossimilhança dos eventos pode ser descrita por duas funções de conjunto: a medida de possibilidade e a medida de necessidade, denotadas respectivamente por  $P$  e  $N$ . Quando  $A$  é uma função de pertinência de um conjunto “crisp” (estritamente clássico)  $A$ , um evento  $B$  é dito possível, se e somente se,  $P(A) > 0$ ; e necessário se,  $N(A) = 1$ ; por definição fazemos  $P(B) = 1$  nas situações respectivas.

## **Modelo da evidência**

A *teoria de evidências* é um dos modelos mais conhecidos para a representação da incerteza em sistemas baseados em conhecimento. Neste modelo, a informação fornecida por uma fonte de conhecimento a respeito do valor real de uma variável  $x$ , definida em um universo de discurso  $X$ , é codificada sob a forma de um *corpo de evidência* sobre  $X$ . Um corpo de evidência é caracterizado por um par  $(F, m)$ , onde  $F$  é

uma família de subconjuntos de  $X$  (isto é,  $\mathcal{A}$ ), e a função de *alocação de massa*  $m$  é uma aplicação de  $2^X$  no intervalo  $[0,1]$ , tal que  $m(A) > 0$ . Cada elemento é chamado *elemento focal*, e  $m(A)$  se refere à evidência relativa a  $A$  unicamente (e não aos subconjuntos de  $A$ ). Nesta teoria, a incerteza ligada ao evento é medida através de duas funções  $e$ , definidas por:

A função de *credibilidade*  $Bel$  mede a que ponto as informações fornecidas por uma fonte sustentam  $A$ . A função de *plausibilidade*  $Pl$  mede a que ponto as informações dadas por uma fonte não contradizem  $A$ . As funções de credibilidade e plausibilidade estão ligadas pela relação:

$$Bel(A) = 1 - Pl(\bar{A}).$$

Isto equivale a dizer que quanto mais se aumenta a evidência sobre uma hipótese, menor se torna a plausibilidade da hipótese contrária.

## 2.4.6. Percepção

A visão, a audição e o tato são as formas de percepção mais estudadas para efeito de incorporação em máquinas. Visão, segundo D. Marr, em 1982, é o processo que permite extrair das imagens do mundo externo uma descrição que seja útil e não poluída com informações irrelevantes. A Visão é dita ativa quando o sistema possui controle sobre o processo de aquisição da imagem, por exemplo através do Zoom, pan, controle de atenção, controle da iluminação especial e outros. O estudo da Visão inclui, portanto, uma série de temas de vanguarda. O processamento de imagens é o processo de melhorar as características das imagens para serem vistas por humanos, sem a preocupação com o que as imagens representam, como a eliminação de distorções, aberrações, filtragem de ruído, correção de brilho, matiz, saturação, ajuste de contraste, entre outros.

Reconhecimento de padrões é o processo de obter, a partir da imagem, a informação de quais objetos estão nela representados, como reconhecimento de caracteres e de impressões digitais.



Computação gráfica é a geração ou modificação de imagens através do uso de computadores. É a Visão, no sentido em que a preocupação aqui é a geração de imagens sintéticas a partir das descrições de cena.

A Visão artificial pressupõe a existência de um dispositivo que permita adquirir imagens. Historicamente, a câmara obscura, já descrita nos manuscritos de Leonardo da Vinci, deu origem à fixação permanente da imagem por J. N. Neipce em 1826 e L. J. M. Daguerre em 1839, com a invenção da fotografia. Em 1923, V. K. Zworykin patenteou um tubo eletrônico que permitia adquirir imagens, o iconoscópio. Em 1928, J. L. Baird demonstrou o princípio da televisão a cores, usando disco de Nipkow. Embora a câmera eletrônica tivesse sido inventada há algum tempo, a necessidade de recursos computacionais limitou o desenvolvimento da visão artificial, até que, em 1963, L. G. Roberts construiu um sistema de visão 3D que compreendia cenas com poliedros. Em 1971, D. A. Huffman desenvolveu uma teoria para desenhos com linhas, posteriormente aperfeiçoado por D. L. Waltz em 1972.

Em 1975, B. K. P. Horn propôs um método para recuperação de forma de objetos a partir de sombreamento. Em 1978, D. Marr e E. Hildred desenvolveram uma eficiente forma de detecção de bordas ( $\nabla G$ ). Em 1979, J. R. Kender apresentou um método para recuperação de forma a partir da textura. Um significativo trabalho sobre estereopsia foi publicado por D. Marr e T. Poggio em 1979. Também nesse ano, D. Marr consolidou a formulação representacional envolvendo esboços primiais, esboços 2 1/2D e Modelo 3D. O estudo das estruturas 3D a partir do fluxo óptico recebeu a devida atenção em 1980, através de Longuet-Higgins e Prazdni, embora já em 1931, Miles tivesse introduzido este tipo de problema.

### **2.4.7. Robótica**

A palavra Robô foi introduzida no contexto de máquinas se assemelhando ao ser humano na peça teatral satírica R. U. R. (Rossum's Universal Robots) de Karel Capek em 1921. Em 1959 a Unimation já lançava o primeiro robô industrial. Em 1962 H. A. Ernst reportava o desenvolvimento de uma mão mecânica com sensores táteis. Em 1960 McCarthy desenvolveu um computador com mãos, olhos e ouvidos. Em 1968 o

problema cinemático de manipuladores controlados por computador foi tratado por Pieper. Hoje em dia, os robôs são universalmente empregados nas mais diferentes tarefas. A robótica é uma ciência interdisciplinar que congrega diversas disciplinas, como mecânica, física de sensores, ciência dos materiais, teoria de controle, eletrônica de potência e muitas outras, incluindo particularmente a IA.

A IA é importante na robótica por possibilitar atividades diversas como planejamento de tarefas e interação com o meio ambiente. Robôs móveis necessitam de capacidades como detecção de obstáculos, geração de trajetórias, análise de cenas e navegação autônoma. Robôs famosos empregados em IA são destacados:

- KAMRO (Karlsruhe Mobile Robot System).
- CMU-Rover (Carnegie Mellon University Rover).
- MOBOT-2 (Mit).
- HILLARE (LAAS Toulouse).
- DAISIE (Distributed Artificially Intelligent System for Interacting with the Environment) da NASA.
- PIRAMIDNET – RN em forma de pirâmide para controle de robôs autônomos desenvolvido pelo L3C da UFSC.

John Von Neumann, o pai do *Autômato Celular*, foi o primeiro a propor um sistema para se reproduzir vida como resultado de regras simples. Seu objetivo era auto reprodução. Um autômato celular é uma cadeia de células que interagem com suas células vizinhas e podem alcançar qualquer dimensão. Cada célula tem seu próprio estado que pode ser uma variável, propriedade ou outra informação.

Recebendo entradas as células conectadas ou mensagens gerais, utiliza seu próprio conjunto de regras para determinar qual seria sua reação. A reação prevista é uma mudança de estado e pode ser também um gatilho para enviar sua própria mensagem. Essas mensagens são passadas para as outras células selecionadas que começam a agir da mesma forma. As células naturais também trabalham desta forma. Um embrião espalha mensagens pela liberação química de células que dizem a este embrião que tipo de célula ele deveria ser. Cada célula tem seu próprio arquivo, o

código DNA para controlar suas ações. As células têm, também, uma planta química e resposta química que atuam como suas interfaces de entrada e saída. Essas plantas são controladas por RNA, os escravos do DNA. Diferentes doses químicas e em diferentes combinações fazem a célula agir de diferentes modos. Às vezes pequenas mudanças como densidade química podem causar radicais mudanças na formação da célula. Desta forma, as barreiras de células são formadas. Por misturas de cooperação e competição, formas complexas, como um bebê, podem ser formadas.

Autômato celular artificial é geralmente bem mais simples, mas pode imitar muitos modelos comportamentais de sistemas biológicos naturais. Cada criatura pode ser como uma célula que tem que competir e cooperar com todas as outras células que usam os recursos disponíveis compartilhados.

“Conway’s Life” é o exemplo mais famoso de autômato celular. Vida é baseada em teorias de Von Neumann, tratando o autômato como uma grade bidimensional em que cada quadro pode estar em dois estados, sendo todas as células idênticas e com as mesmas regras, como segue:

- Uma célula viva com zero ou um vizinho vivo morre por isolamento;
- Uma célula viva com quatro ou mais vizinhos vivos morre por sufocamento;
- Uma célula morta com exatamente três vizinhos vivos se torna viva;
- Todas as outras células mantêm seu estado.

Para Almeida, 1999, um autômato pode ser considerado como uma particularização de um sistema dinâmico. Informalmente, pode-se dizer que o rótulo “dinâmico” tem o mesmo significado de “casual”: as entradas passadas influenciam o futuro mas o contrário não é verdadeiro. Ou seja, a noção matemática de sistema dinâmico serve para descrever o fluxo de causa entre o passado e o futuro.

Ao se utilizarem os recursos de Hipermídia ou modelo formal de Hipertexto, Pagano, 89 e Pagano, 90, relaciona a gerência por computador de vários meios de apresentação de informação, fundamentados na teoria dos autômatos, incorporando não somente os aspectos declarativos, mas também os aspectos dinâmicos, onde o autômato responde a qualquer seqüência de entrada de dados.

# 3

## INSTRUMENTAÇÃO, CONTROLE E AUTOMAÇÃO

### 3.1. Conceitos fundamentais

Entende-se por *Sistema*, um dado conjunto de princípios de controle e automação, uma parte do Universo sobre a qual fixa-se a atenção do projetista. Um sistema é constituído, em geral, de diversos componentes que interagem entre si, em conformidade com as leis da Natureza, (Nascimento, 2000). As grandezas envolvidas nas interações entre partes do sistema podem ou não ser ajustadas de acordo com as especificações de um projetista. Quando o controle é realizado com pouca ou nenhuma intervenção humana, diz-se que é do tipo automático. Nesse contexto, automatizar é dotar os sistemas de mecanismos ou dispositivos que, com um mínimo de intervenção humana, permita que sejam alcançadas as especificações de segurança, produtividade, qualidade, conforto e outras.

### 3.2. Problemas de controle

O controle visa estabelecer uma estratégia de atuação sobre um sistema, de modo que este se comporte de forma conveniente. A especificação do comportamento desejado pode envolver conceitos como estabilidade, rejeição de distúrbio, robustez a incertezas no modelo, forma da resposta do sistema a entradas padrão, simplicidade de implementação, custo de operação, entre outras.

Os modelos matemáticos são representações úteis e necessárias do sistema em estudo, de forma tal que um sistema elétrico possa representar um modelo de um circuito eletrônico.

Uma classe de problemas de controle muito encontrada na prática é a de manter próxima a um valor de referência  $r$ , uma grandeza de saída  $y$ , por exemplo o nível de líquido em um tanque, através da manipulação de uma variável de controle  $u$ , que

poderia ser, como na ilustração, a velocidade de rotação do motor acoplado a uma bomba hidráulica.

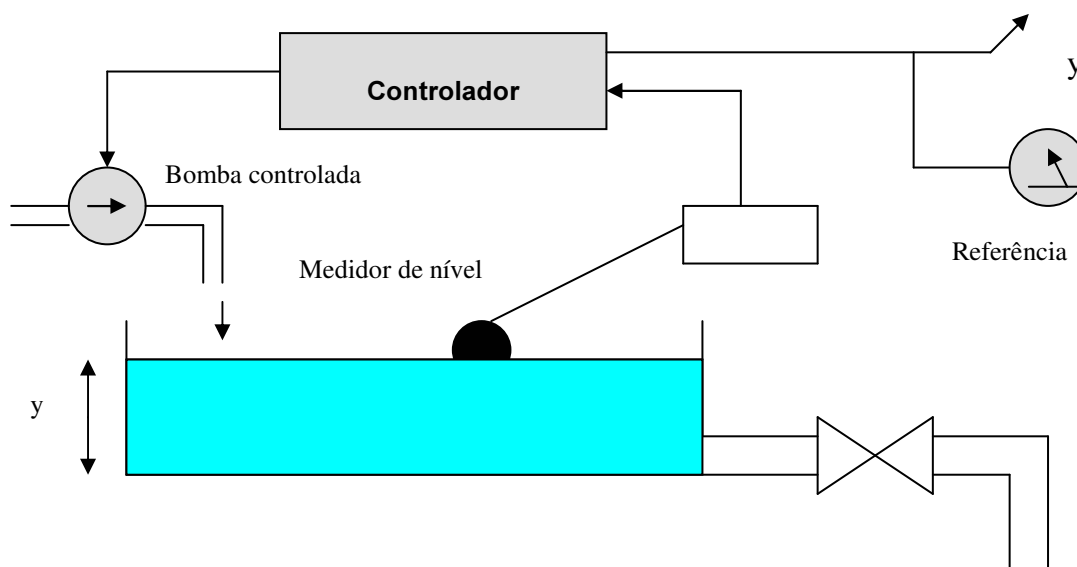


Fig. 3.1. Sistema de controle de nível de líquido em um tanque

Supondo que existam sensores que fornecem a leitura da saída  $y$ , o objetivo é projetar um equipamento que construa o sinal  $u$  a partir do sinal de erro  $e = r - y$ . O sinal de erro é a diferença entre o que se deseja  $r$  e as saídas de fato  $y$ . apenas por facilidade de notação, assume-se que  $t \in \mathbb{R}_+$ ,  $r(t) \in \mathbb{R}$ ,  $u(t) \in \mathbb{R}$  e  $e(t) \in \mathbb{R}$ . Os sinais  $n(t)$  e  $w(t)$  são perturbações externas.

Um problema de controle bem formulado deve incluir um modelo matemático qualitativo ou quantitativo do sistema e as especificações desejadas. Segundo Nascimento em 2000, as boas soluções incluem:

### **Análise de solvabilidade:**

Consiste em verificar se o problema de controle automático admite ou não uma solução. Muitas vezes as especificações de desempenho são tais que o problema é insolúvel. Principais problemas de formulação:

- Minimização de funções que não possuem pontos de mínimo;
- Estabilização de sistemas cuja parte não-estável é não-controlável;
- Estimação de estados de sistemas não-observáveis;
- Controle de sistemas com taxas de amostragem inferior às frequências de interesse

**Metodologia de projeto:**

Identificar que existe pelo menos uma solução, uma metodologia deve ser adotada, visando eficácia nos procedimentos onde métodos numéricos podem estar integrados em pacotes computacionais que permitam redução de tempo e esforço. Métodos heurísticos podem envolver técnicas tipo tentativa-e-erro ou até mesmo Sistemas especialistas.

# 4

## LÓGICA E DEDUÇÃO

Muitas características dos computadores digitais empregados em tarefas podem ser ressaltadas, como as funções complexas de controle exigindo seleção automática de sensores ativos assim como a facilidade de alteração dos controladores, permitindo versatilidade na adaptação e incorporação de novos elementos ao sistema. A portabilidade e a confiabilidade dos sistemas interferem diretamente na qualidade do processamento, trazendo eficácia na interação homem-máquina, podendo apresentar resultados notáveis no processo de provimento de melhorias, no treinamento e no aprendizado dos sistemas lógicos.

### 4.1. Sistemas discretos

Computadores digitais utilizados em controle automático recebem excitação dos sensores, em determinados instantes de tempo e, após processá-los, produzem saídas que são aplicadas nos transdutores. As entradas são freqüentemente sinais analógicos, convertidas em sinais digitais e as saídas transformadas em grandezas analógicas novamente. Inevitavelmente, durante o processo, muitos dados se apresentam no formato numérico e, quando disponibilizados em intervalos de tempo igualmente espaçados, serão modelados como seqüenciais.

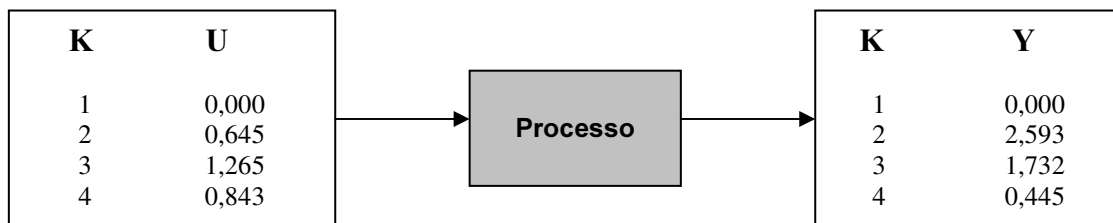


Fig. 4.1. Representação de um sistema discreto

A dinâmica pode ser expressa na forma de equações:

$$Y_k = -a_1 Y_{k-1} - a_2 Y_{k-2} - \dots - a_n Y_{k-n} + -b_0 U_k - b_1 Y_{k-1} - \dots - b_m Y_{k-m}$$

Onde  $a_i$  e  $b_j$  são constantes e as seqüências  $\{U_k\}_{k = \dots, -2, -1, 0, 1, 2, \dots}$  e  $\{Y_k\}_{k = -2, -1, 0, 1, 2, \dots}$  são respectivamente, a entrada e a saída. Quando  $a_n \neq 0$ , o modelo é dito ser de ordem  $n$ .

Amostrar é focalizar a atenção sobre uma parte de um todo. Assim, amostrar um sinal  $w(t)$ , suposto definido para  $\forall t \in \mathbb{R}$ , é obter informações sobre o valor de  $w(\cdot)$  em alguns instantes  $t_1 + t_2 + \dots + t_n$ , muitas vezes supostos distribuídos uniformemente na forma  $t_k = KT$ , onde  $K \in \mathbb{Z}$  e  $T$  é um número real constante, chamado de período de amostragem.

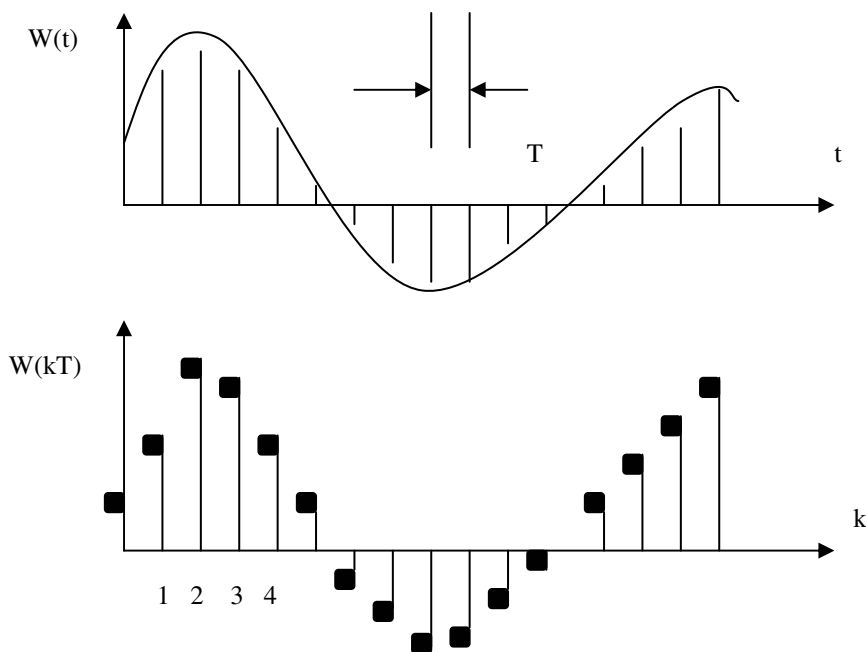


Fig. 4.2. Amostra do sinal  $W(t)$  em instantes diversos



## 4.2. Cálculo sentencial

Ao se verificar a veracidade de sentenças complexas, ou proposições ou fórmulas a partir da veracidade ou falsidade de suas partes, a metodologia dos cálculos sentenciais se apresenta a partir da combinação das sentenças mais simples e se impõem através do uso de conectivos.

A veracidade que se deseja estabelecer é quanto à legitimidade dos argumentos, utilizando regras elementares e se concluir definitivamente.

Considerando os argumentos:

- |    |   |     |   |
|----|---|-----|---|
| i) | <ul style="list-style-type: none"> <li>- Ave põe ovos</li> <li>- Águia é ave</li> <li>- Águia põe ovos</li> </ul> | ii) | <ul style="list-style-type: none"> <li>- Ave põe ovos</li> <li>- Mickey é ave</li> <li>- Mickey põe ovos</li> </ul> |
|----|---|-----|---|

Ambos os argumentos estão corretos. A verificação de se *Mickey é ave* não é o problema aqui tratado. Eventualmente, Mickey poderia ser o nome de um papagaio. Mesmo uma sentença aparentemente óbvia como *Mickey é camundongo* poderia ser questionada, uma vez que dependendo do contexto, o significado pretendido poderia ser *Mickey é desenho*. Assim, a investigação dos critérios para estabelecimento da verdade está além do escopo de nossas afirmativas.

É importante observar que o uso de símbolos iguais para objetos diferentes pode resultar em conclusões surpreendentes. Assim, usando-se o símbolo comum a duas verdades podemos ter designações e conclusões das mais diversas em relação ao esperado. Este fato é bem conhecido a todos aqueles que estão familiarizados com os pontos fundamentais da semiologia geral.

Os conectivos sentenciais usualmente empregados são:

$\wedge$ E	$\vee$ OU	$\rightarrow$ IMPLICA	$\leftrightarrow$ EQUIVALE	$\neg$ NÃO
------------	-----------	-----------------------	----------------------------	------------

Tab. 4.3. Conectivos sentenciais

É importante observar também a possibilidade de dispor dois conectivos combinados e que a veracidade ou não pode ser aferida com o uso de tabelas verdade, como as que seguem:

<b>A</b>	<b>b</b>	<b>a ∧ b</b>	<b>a ∨ b</b>	<b>a → b</b>	<b>a ↔ b</b>	<b>¬a</b>
V	V	V	V	V	V	F
V	F	F	V	F	F	F
F	V	F	V	V	F	V
F	F	F	F	V	V	V

Tab. 4.4. Tabela verdade dos conectivos combinados

E para verificação, por exemplo, da veracidade de:  $a \rightarrow (b \vee c)$ , pode-se construir a Tabela Verdade abaixo:

<b>a</b>	<b>b</b>	<b>c</b>	<b>b ∨ c</b>	<b>a → (b ∨ c)</b>
V	V	V	V	V
V	V	F	V	V
V	F	V	V	V
V	F	F	F	F
F	V	V	V	V
F	V	F	V	V
F	F	V	V	V
F	F	F	F	V

Tab. 4.5. Tabela verdade dos conectivos com base em uma veracidade

Na situação da tabela 4.5, nota-se que a expressão  $a \rightarrow (b \vee c)$  não assume o valor V para quaisquer valores de a,b e c. Entretanto, existem expressões para as quais o valor é sempre V, independentemente dos valor dos elementos.

Essas expressões são denominadas Tautologias. Algumas importantes são destacadas a seguir:

$\mathbf{a \rightarrow a}$	Identidade	$\mathbf{\neg(\neg a) \leftrightarrow a}$	Dupla negação
$\mathbf{a \leftrightarrow a}$	Equivalência	$\mathbf{(a \vee (\neg a))}$	Terceiro excluído
$\mathbf{(a \wedge a) \leftrightarrow a}$	Idempotência	$\mathbf{\neg(a \wedge (\neg a))}$	Não contradição
$\mathbf{(a \vee a) \leftrightarrow a}$	Idempotência	$\mathbf{\neg a \rightarrow (a \rightarrow b)}$	Negação do antecedente

Tab. 4.6. Algumas expressões tautológicas

A Dedução é a obtenção de uma conclusão verdadeira a partir de premissas consideradas verdadeiras e de axiomas válidos.

Em princípio a dedução pode ser verificada através do uso de tabelas verdade, onde para denotar que os axiomas  $a_1 - a_m$  permitem deduzir  $b$ , a partir das premissas  $p_1 - p_n$  é usada a grafia:  $p_1, \dots, p_n, a_1, \dots, a_m \Rightarrow b$

Como exemplo, consideremos a verificação da legitimidade da dedução:  $\mathbf{a \rightarrow b, \neg b \Rightarrow \neg a}$

Como um bom exemplo, atribuem-se significados para os seguintes itens:

A = Tração excessiva no fio

B = Fio se rompe, ou seja:

“Se (tração excessiva) Então (Fio se rompe)”, “Não é verdade que (fio se rompeu)”, de onde se deduz que “Não é verdade que (houve tração excessiva no fio)”. A legitimidade dessa dedução pode ser verificada pelas seguinte tabela verdade:

<b>a</b>	<b>b</b>	<b>¬b</b>	<b>a→b</b>	<b>¬a</b>
V	V	F	V	F
V	F	V	F	F
F	V	F	V	V
F	F	V	V	V

Tab. 4.7. Legitimidade da dedução do problema “Tração excessiva”

Observa-se na tabela 4.7, quando  $a \rightarrow b$  e  $\neg b$  são simultaneamente V, tem-se que o valor de  $\neg a$  é V.

Uma forma de se reduzir o emprego de tabelas verdade de grandes dimensões é utilizar algumas formas já conhecidas de dedução, como as sugeridas abaixo:

1. Modus Ponens  $a, a \rightarrow b \Rightarrow b$
2. Modus Tollens  $a \rightarrow b, \neg b \Rightarrow \neg a$
3. Silogismo Hipotético  $a \rightarrow b, b \rightarrow c \Rightarrow a \rightarrow c$
4. Silogismo Disjuntivo  $\neg a, a \vee b \Rightarrow b$
5. Dilema Construtivo  $a \vee c, (a \rightarrow b) \wedge (c \rightarrow d) \Rightarrow b \wedge d$
6. Dilema Dedutivo  $((\neg b) \vee (\neg d)), (a \rightarrow b) \wedge (c \rightarrow d) \Rightarrow (\neg a) \vee (\neg c)$
7. Simplificação  $a \vee b \Rightarrow a$
8. Conjunção  $a, b \Rightarrow a \vee b$
9. Adição  $a \Rightarrow a \vee b$
10. Dedução por absurdo  $\neg a, a \Rightarrow b$

### 4.3. Cálculo de predicados

O cálculo de predicados também serve para estudar a legitimidade ou não de sentenças, mas inclui os efeitos dos quantificadores  $\forall$  e  $\exists$  e ainda os predicados expressos por verbos e objetos.

Considerando a dedução:

(todas as crianças estudam)  
(Alice é uma criança)  
(Alice estuda)

Que pode ser representada simbolicamente por:

$$\{(\forall x [\text{Criança}(x) \rightarrow \text{Estuda}(x)]) \wedge (\text{Criança}(\text{Alice}) = V)\} \Rightarrow \text{Estuda}(\text{Alice}) = V$$

A importância do quantificador pode ser facilmente percebida examinando-se a mesma estrutura anterior, mas com  $\exists$  no lugar de  $\forall$ :

Considerando a nova dedução:

(existem as crianças que estudam)  
(Alice é uma criança)  
(Alice estuda)

Que pode ser representada simbolicamente por:

$$\{(\exists x [\text{Criança}(x) \wedge \text{Estuda}(x)]) \wedge (\text{Criança}(\text{Alice}) = V)\} \Rightarrow \text{Estuda}(\text{Alice}) = V$$

Obviamente a Segunda dedução é incorreta.

O proceso de dedução no cálculo de predicados é similar ao cálculo sentencial, exceto por pasos adicionais para eliminação ou introdução de quantificadores. A eliminação de  $\forall$  recebe o nome de Instanciação Universal e a introdução de  $\exists$  de Generalização Existencial.

# 5

## REPRESENTAÇÃO DE CONHECIMENTO

Intuitivamente, conhecimento é um conjunto de informações que permite articular os conceitos, os juízos e o raciocínio, usualmente disponíveis em um particular domínio de atuação. Ao se empregar um sistema de símbolos, esse conhecimento pode ser representado de modo a permitir atividades como inferência ou memorização.

A representação pode ser feita de múltiplas formas, tais como através de textos, fórmulas matemáticas, figuras, maquetes, filmes, discos protótipos e muitas outras alternativas. Um texto específico pode ser composto de sentenças do cálculo de predicados ou ainda, esse texto pode estar na forma binária em memória de computador, para processamento digital. Alternativamente, o conhecimento pode estar embutido nos valores de determinados componentes de um circuito, como o caso de redes neurais.

### 5.1. Representações lógicas

Uma opção muito comum para representar conhecimentos heurísticos é usar regras de produção, tipicamente na forma “Se (condições) Então (conclusões)”. Embora regras de produção sejam apenas casos particulares de sentenças tratáveis pela Lógica de Predicados, muitos sistemas práticos empregam essa estrutura de representação. Muitos sistemas especialistas utilizam regras de produção aliadas a uma máquina de inferência para realizar a sua tarefa.

**Se** (produz\_leite  $\wedge$  tem\_pêlos)

**Então** (mamíferos)

**Se** (mamífero  $\wedge$  come\_carne)

**Então** (carnívoro)

**Se** (mamífero  $\wedge$  possui\_presas  $\wedge$  possui\_garras)

**Então** (carnívoro)

**Se** (mamífero  $\wedge$  possui\_casco)

**Então** (unglado)

**Se** (carnívoro  $\wedge$  pardo  $\wedge$  pintado)

**Então** (onça)

**Se** (carnívoro  $\wedge$  pardo  $\wedge$  listrado)

**Então** (tigre)

À partir de conhecimentos sobre as características dos objetos, verifica-se que as conclusões são obtidas pelo encadeamento progressivo. Uma forma de usar os conhecimentos na forma de regras de produção é partir das conclusões que poderão ser provadas ou não, conjecturando que características pré-conhecidas são infusas na verificação se as condições são satisfeitas para se obterem as respostas conclusivas.

## 5.2. Representações estruturadas

A rede semântica é um tipo de representação de conhecimento estruturada, onde uma coleção de nós conectados por arcos. Os nós representam objetos, conceitos ou eventos e os arcos representam as relações. Usualmente os nós e os arcos são etiquetados para indicar o tipo de conhecimento que representam:

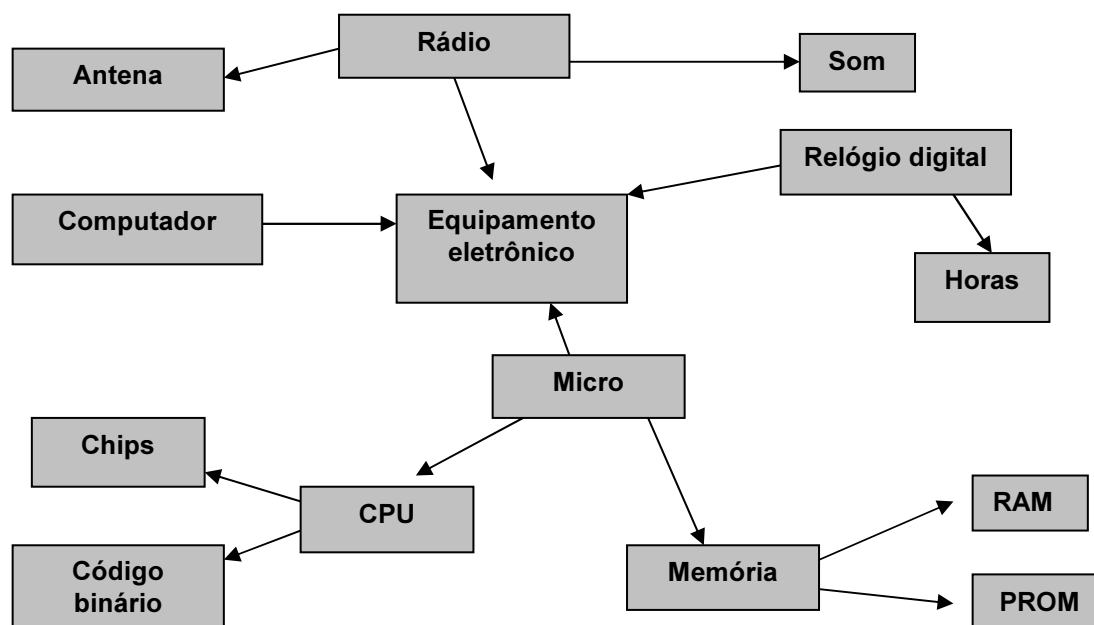


Fig. 5.1. Rede semântica representando equipamentos eletrônicos

### 5.3. Métodos de busca

Problemas de IA podem utilizar métodos de busca para o processo de solução de problemas lógicos. A busca é um processo onde se realizam acessos nos sentidos de encadeamento progressivo ou retroativo. A vantagem de um ou outro sentido depende da particular estrutura a ser pesquisada.

O desenrolar de um jogo da velha pode ser caracterizado por árvores e para que se possa selecionar opções para o jogo, há necessidade de uma heurística que permita fornecer a cada jogada, um índice que reflita a vantagem posicional alcançada pelo movimento escolhido. Em cada situação é contada a diferença entre o número de alinhamentos possíveis para cada jogador. Se considerarmos cada posição situada na  $i$ -ésima linha e na  $j$ -ésima coluna, a denotação  $(i,j)$ . Nestas condições, conforme a figura abaixo, o símbolo (O) na posição  $(1,2)$  e um símbolo (X) na posição  $(2,2)$ , implica que o jogador que utiliza o símbolo (O) pode obter 4 alinhamentos enquanto o que o utiliza (X) consegue 6 alinhamentos.

Portanto, o jogador (X) atribui um valor  $+2 = +6 - 4$  para a situação.

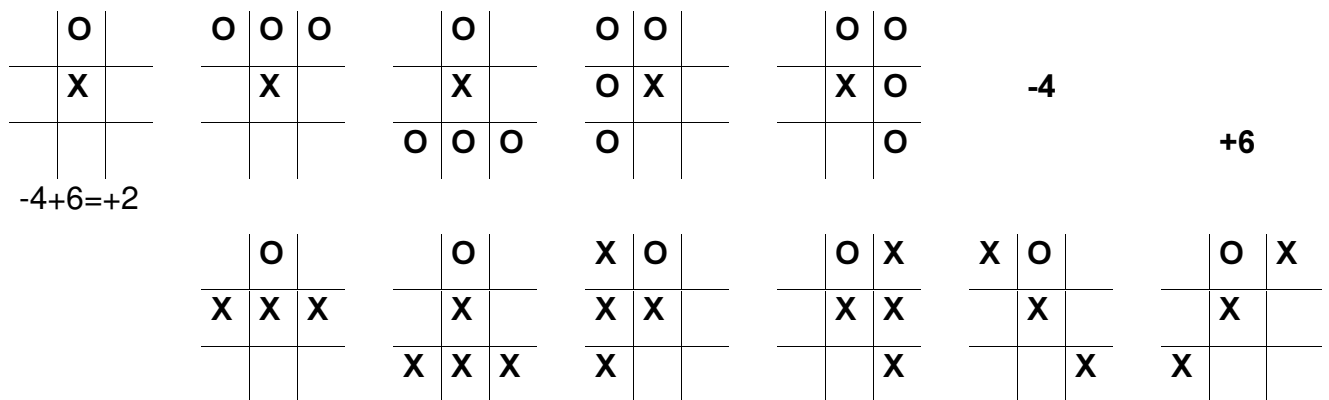


Fig. 5.2. Alinhamentos possíveis para o jogador (O)=(4)

Alinhamentos possíveis para o jogador (X)=(6)

A diferença  $-4+6=+2$  é o índice para a configuração

No mecanismo apresentado, caso o jogador (X) se posicione em  $(1,1)$  e considerando a diferença entre o número de alinhamentos possíveis para cada jogador,



o jogador (O) poderá obter valor  $-1$  pela ocupação de  $(2,2)$  que é a situação mais favorável para (O). Se o jogador (O) ocupar a posição  $(1,2)$  ele obterá o valor  $+1$ , ou seja, pior para (O). Notar que o melhor caso para (O) será também o pior caso para (X).

Assim, considerando-se as possibilidades para a primeira jogada de (X), os valores associados são:

$$(X) \text{ em } (1,1) \rightarrow \text{valor } -1, \text{ pois } -1 = \min\{-1,0,1\}$$

$$(X) \text{ em } (2,2) \rightarrow \text{valor } +1, \text{ pois } 1 = \min\{1,2\}$$

$$(X) \text{ em } (2,1) \rightarrow \text{valor } -2, \text{ pois } -2 = \min\{-2,-1,0\}$$

Considerando o melhor dos piores casos, o jogador (X) deve optar por  $(2,2)$ , que corresponde ao máximo entre  $-1$ ,  $+1$  e  $-2$ . Por outro lado, o valor  $-1$  corresponde ao mínimo entre  $1$ ,  $0$ ,  $1$ ,  $0$  e  $-1$  da sub-árvore à esquerda. Analogamente, o valor  $-2$  corresponde ao mínimo entre  $-1$ ,  $-2$ ,  $0$ ,  $0$  e  $-1$  da sub-árvore à direita. Essa estratégia de decisão é chamada de MAX-MIN, ou seja, maximizar o ganho considerando os piores casos obtidos ao se calcular o máximo dos mínimos.

# 6

## SISTEMAS DE INFORMAÇÃO INTELIGENTES

Sistemas Especialistas, utilizando técnicas de inteligência artificial numa base de conhecimento e programa de computador que usa conhecimento e procedimento de inferência podem ser desenvolvidos para resolver problemas. A base de conhecimento consiste de fatos e heurística, onde "fatos" constituem o corpo da informação que é de notório conhecimento científico, público e geralmente de acordo com o pensamento de peritos da área e "heurística" (na maior parte confidencial), que são regras de raciocínio plausível, regras de bom senso características de uma decisão no nível-especialista.

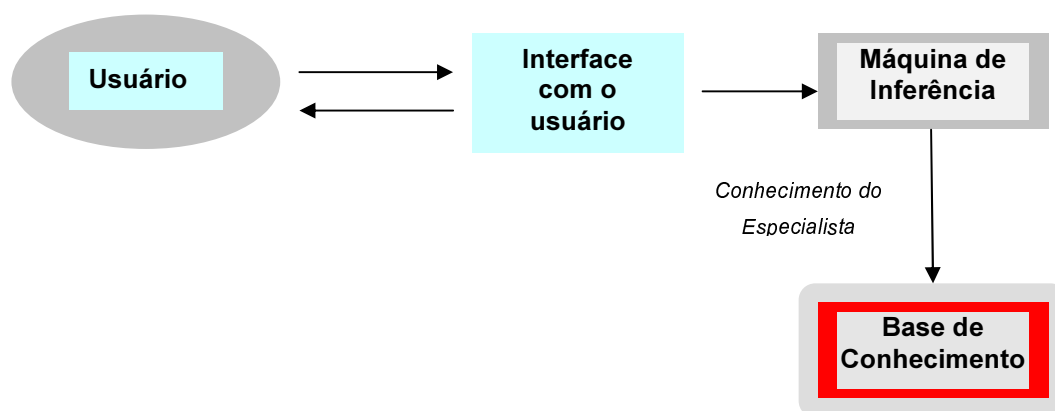


Fig. 6.1. Estrutura convencional de Sistema Especialista

### 6.1. A engenharia de conhecimento

Basicamente, de acordo com Feigenbaum (1988), 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, podem ser considerados modelos da perícia pelos 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 pouco 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.

Para que um sistema seja considerado especialista, Farreny (1985) coloca que alguns componentes são essenciais à sua caracterização, são eles:

- 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 susceptível a mudanças.

Os SE são programas de IA que capacitam um computador a auxiliar o usuário durante um processo de tomada de decisão. O know-how do perito humano é utilizado para instruir o computador a resolver um problema ou a tomar uma decisão. A máquina pode, então, auxiliar ou aconselhar outros usuários na resolução do mesmo problema através:

- do conhecimento especializado do usuário humano;
- da habilidade do computador para armazenar grande quantidade de textos e dados e considerar todas as possibilidades a alta velocidade.

Segundo Carvalho, 2001, os computadores são máquinas que tentam reproduzir a capacidade de articulação mental que o homem tem para reunir dados, interpretá-los, contextualizá-los e transmiti-los, sob a forma de informações. Por isso, guardam em seus interiores sistemas de informações que nada mais são do que combinações de dados que, estruturadas, são capazes de produzir a informação desejada.

Assim, Chorafas, 1988 define: "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."

Os SE são também conhecidos por sistemas cognitivos e a tecnologia utilizada para seu desenvolvimento é chamada de engenharia de conhecimento. Alguns autores costumam chamar de sistemas cognitivos apenas os sistemas de pequeno porte.

Jacques Quibel, em 1989, inicia sua definição da engenharia do conhecimento pela definição da própria palavra engenharia: "A engenharia comporta os métodos para estudo de um projeto e realização de um trabalho complexo por equipes diversas, de especialidades complementares." A partir daí ele continua afirmando que "desenvolver um SE é aplicar uma "engenharia" do conhecimento, a construção de uma informática lógica, refere-se a fatos e idéias e a capacidade de tratá-los, para produzir um diagnóstico, uma preconização, ou, de fato, uma decisão."

Segundo Barreto, 2000, a partir de uma base de dados só se pode extrair o que se introduziu e a qualidade das respostas depende somente da qualidade dos dados de entrada. O computador serve para memorizar e extrair dados e com estes dados vem as informações que estão associadas a eles. A palavra *dados* embora sendo familiar a todo profissional de computação por se apresentar em sua estrutura em forma de árvores, pilhas, filas, etc., freqüentemente assume condições um pouco diferenciadas, uma vez que para utilização eficiente, passa a ser objeto de manipulações sintáticas.

Estruturalmente, um SI inteligente, S pode ser definido como:

$$S = \langle D, A, F, M, R, V \rangle$$

Sendo: D: Conjunto de dados

A: Atributos considerados relevantes e disponíveis

F: Dados provenientes de uma base de dados

M: Memória onde se armazenam os dados

R: Regras de combinação de dados

V: Conjunto de visões sobre dados manipulados

## 6.2. A extração de conhecimento

O primeiro passo para a extração de conhecimento é a leitura da BD pelo sistema a fim de construir um Dicionário com as palavras mais relevantes. Durante esta leitura o sistema controí módulos para representar cada palavra diferente encontrada na BD. Estes módulos são compostos por um sub-módulo indexador denominado germe e vários outros sub-módulos complementares denominados halos. As combinações de um germe e seus halos representam as palavras encontradas na BD, que possuem a mesma propriedade de indexação, como por exemplo, as palavras associadas a diferentes tempos verbais e que são compostas pelo radical verbal e as terminações de sua conjugação.

O passo seguinte, impõe ao usuário manusear o dicionário, a fim de eliminar as palavras que nada significam para o propósito da análise, definir as relações de sinonímia entre as palavras remanescentes, etc. Estas palavras, assim agrupadas devem descrever conceitos primitivos pertinentes à análise a ser implementada. Este dicionário é chamado, aqui, Dicionário de Conceitos Primitivos, DCP.

Uma vez definido o DCP, o usuário deve estabelecer a sintaxe que será responsável pela associação de palavras para formação de frases. Esta sintaxe deve descrever as relações de interesse que possam se estabelecer entre os conceitos primitivos.

Em seguida, o usuário associa classes da sintaxe assim definida, às palavras escolhidas no DCP, para gerar o Dicionário de Termos (DT) a ser utilizado para análise das frases na BD.

Na próxima etapa, o sistema usa o DT assim constituído, para extrair as combinações de palavras mais freqüentes que satisfaçam a sintaxe definida. É criado então um Dicionário de Frases (DF). Estas frases podem representar conceitos complexos existentes na BD, uma vez que a associação de conceitos primitivos (palavras) leva à representação de idéias (frases) mais elaboradas.

Um processo recursivo, com a repetição de passos, permite ao usuário analisar combinações de frases na BD, na tentativa de analisar conhecimentos recursivamente mais complexos. Esta recursividade é estabelecida, quando a BD é reescrita apenas com as frases do DF. Este processo recodifica a BD através da substituição, no proceso de reescritura, das frases da BD que sejam idênticas às do dicionário, por um código interno associado a estas frases. Abaixo, o modelo inspirado por Fayyad, em 1996:

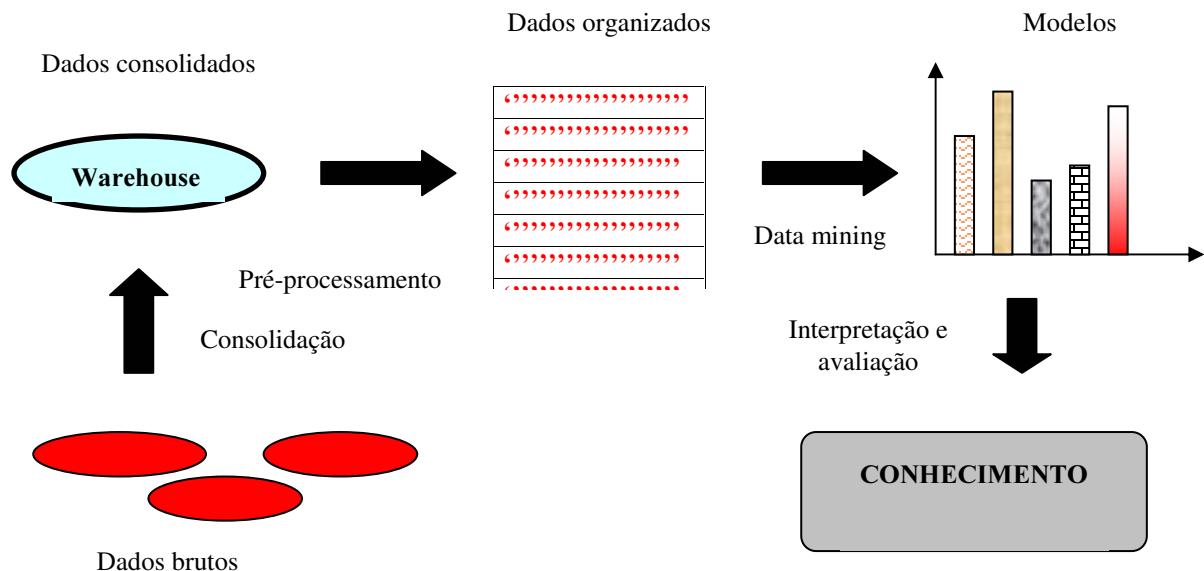


Fig. 6.2. Processo completo de extração de conhecimento em base de dados

Para montar as bases de conhecimento, são montadas as regras com um primeiro formalismo para expressar o conhecimento de uma maneira simbólica. Segundo Pellegrini, 1995, as regras têm a vantagem de ter um formalismo simples, uniforme, transparente e fácil de fazer inferências. As regras podem ser elicitadas diretamente do domínio dos especialistas e expressas através do grau de abstração do especialista, metodologia esta que pode apresentar alguma inconsistência porque dependeria diretamente do conhecimento do especialista sobre a organização de regras de forma hierarquizada.

Pode-se estabelecer um comparativo, segundo Barreto, 2000, entre as principais características de um SI tradicional e o Data Mining:

***** SI Tradicional *****	***** Data Mining *****
Foco na manipulação de informação	Foco na escolha dos dados
Foco no projeto e programação do sistema	Foco em busca e representação
30% do esforço em análise e projeto	70% do esforço na preparação dos dados
70% do esforço na programação e teste	30% na geração do modelo e teste
Prototipagem cara	Prototipagem barata
Desenvolvimento pouco interativo	Desenvolvimento interativo
Manutenção com intervenção humana	Manutenção exige reaprendizado

Tab. 6.3. Comparativo SI tradicional e Data mining

# 7

## O MAPEAMENTO COGNITIVO CEREBRAL

Um sistema foi utilizado para estudar a atividade cerebral relacionada a várias atividades cognitivas em grupos de adultos e crianças normais e crianças excepcionais. Os resultados desses mapeamentos na população de aprendizes excepcionais podem ser confrontados com os achados de um estudo de Ressonância Magnética Estrutural, (Morais, 1998).

Muitas são as causas das disfunções cerebrais que complicam o processo de aprendizagem das pessoas. Qualquer tentativa de melhorar este processo, requer uma boa compreensão da funcionalidade de cada um desses cérebros, de modo a:

- Identificar claramente as potencialidades e dificuldades de cada indivíduo;
- Definir eficientemente as melhores estratégias de ensino que garantam o desenvolvimento cognitivo do indivíduo.

É um ambiente computacional que fornece as ferramentas necessárias para desenvolver melhores técnicas, tanto para diagnóstico quanto ensino de indivíduos. Na prática, mostrando a sua eficiência como ferramenta auxiliar na educação, foi desenvolvido por Tony Buzan enunciando um poderoso conceito que mescla as habilidades naturais e os recursos tecnológicos, denominado MIND MAPPER.

A mais essencial habilidade de receber informação e expressar seus pensamentos depende da sua habilidade de extrair palavras essenciais e analisar seus próprios co-relacionamentos entre si, através de montagem daquelas palavras chaves ou idéias principais em cima de um Mapa mental (Mind Mapper).



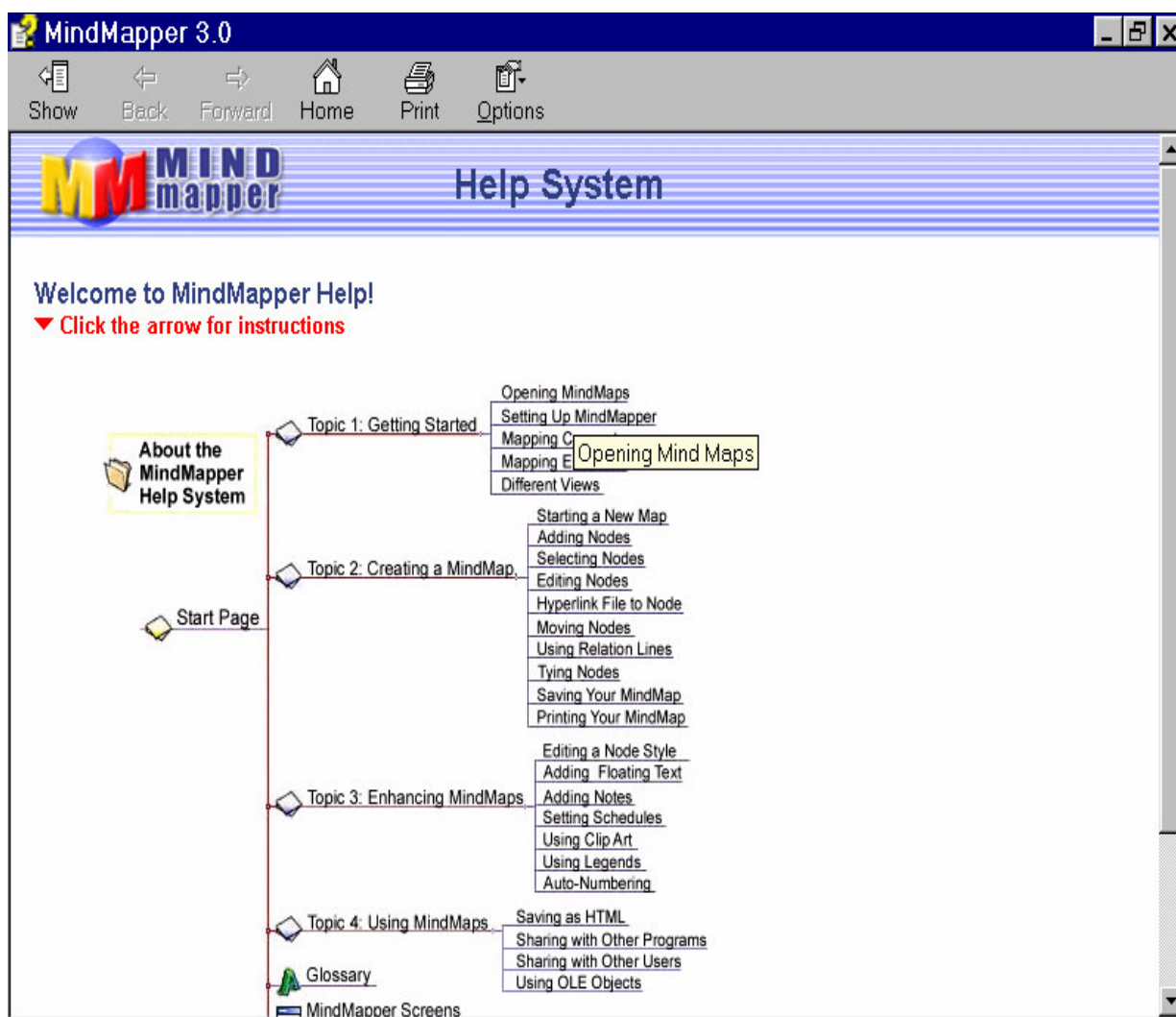


Fig. 7.1. Organização funcional do Mind Mapper

A habilidade para organizar informações de muitas fontes em uma ordem lógica se tornou o fator mais importante para a sobrevivência no mundo da informação. O cérebro humano está dividido em dois hemisférios, onde cada um, pela sua própria natureza pode processar informação que inicialmente parece não ter um padrão ou ordem definidos. Porém, a capacidade de processar informação visual se apresenta de forma muito mais eficaz aos olhos do usuário.

O método de visualizar partes críticas dos pensamentos numa forma genealógica, parecida com uma teia de aranha, ajuda na correlação de idéias pertinentes e correlatas, trazendo muitos benefícios em muitas aplicações científicas.

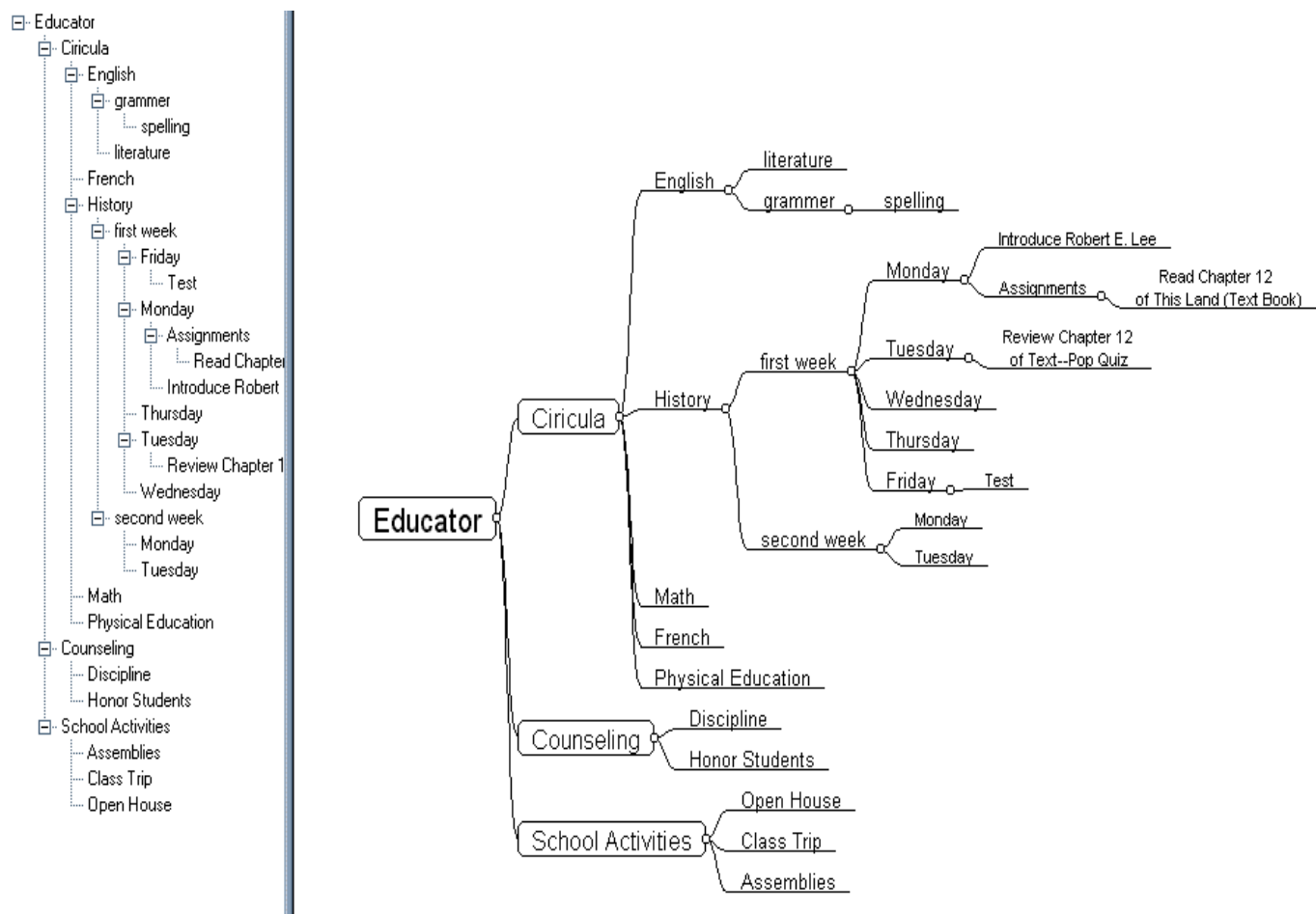


Fig. 7.2. Exemplo de um Mind Mapper utilizado em aprendizagem

Muitos modelos se apresentam com as mais diversas finalidades:

- Definição de metas
- Classificação e categorias
- Genealogia
- Organização de prioridades
- Administração de projetos
- Metas pessoais
- Educação gráfica
- Representação de conceitos
- Fabricação industrial

Muitos softwares com a função específica se encontram disponíveis e alguns dos mais comuns se destacam nos itens a seguir:

## 7.1. Mind Mapper 3.2

Composto por “Topics” (idéia central) e “Branches” (ligações para execução de tarefas específicas), a idéia central é ilustrada com os pontos principais dos eventos que se deseja construir, e muitas vezes se apresentam dispostos em níveis. Os níveis de “Branch” em torno do “Topic” não possuem limites de quantidade nem de disposição, a não ser pela capacidade de memória do computador onde se processam.

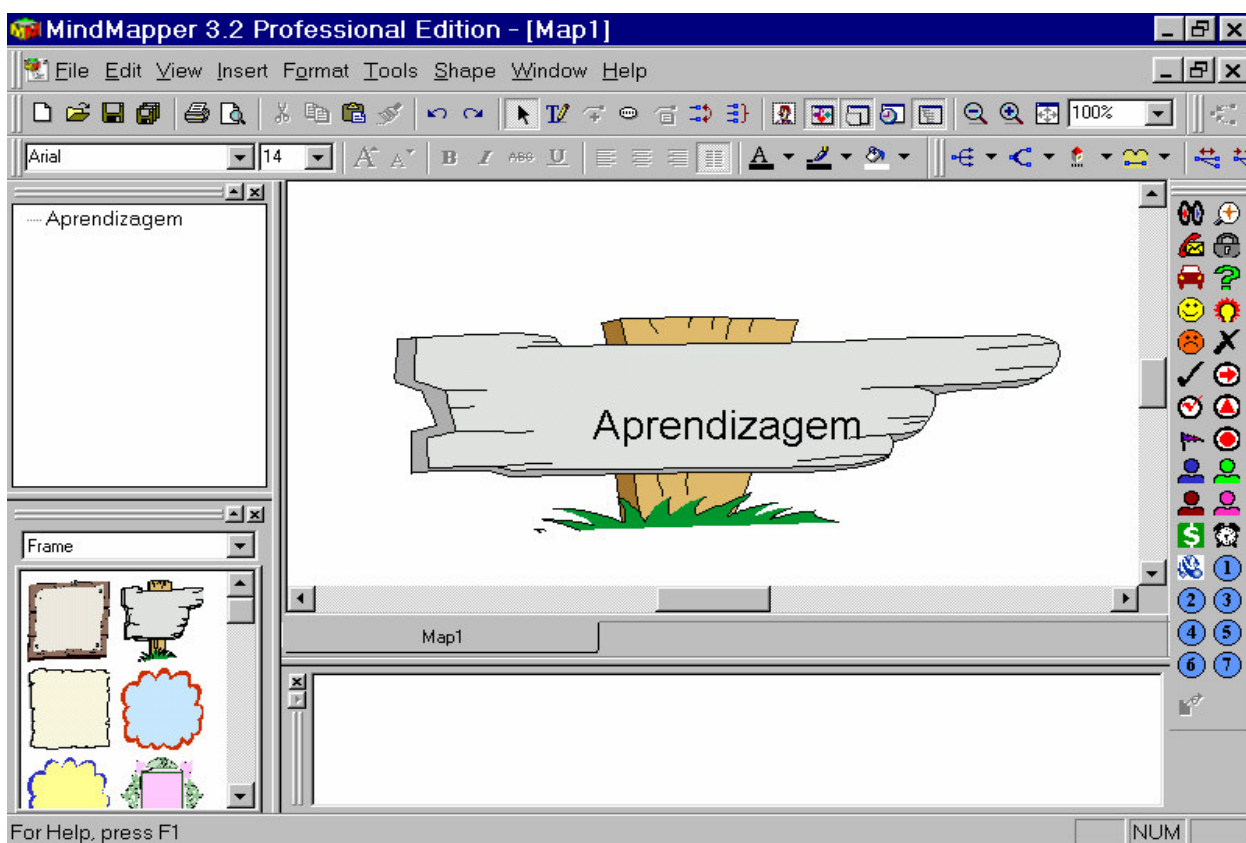


Fig. 7.3. Tela inicial de apresentação do Mind Mapper 3.2.

Este software apresenta muitas características interessantes, especialmente na disposição radial das atividades componentes do “Branch”, indicando, na maioria das vezes, o grau de importância dentro do que o mapa propõe.

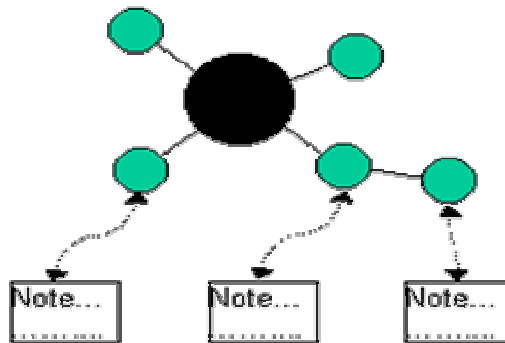


Fig. 7.4. Disposição radial dos “Branches” e dos “Topics”



Fig. 7.5. Disposição radial dos “Branches” e dos “Topics” na própria tela

O Mind Mapper Se apresenta em vários formatos, como segue:

- MAP** Utilizado diretamente em Mind Mapper
- TXT** Utilizado pelo processador Word
- RTF** Utilizado diretamente pelo Mind Mapper
- EMF** Utilizado por softwares em forma de figura
- HTML** Utilizado para criar o Internet Web Site

## 7.2. Mind Mapper Jr

Desenvolvido com o objetivo de transformar idéias e conceitos em imagens e, baseado nas teorias do mapeamento mental original, permite que o usuário ilustre de forma bastante simples, gravuras em quadros, ícones, molduras e comentários. Apenas teclando em gravuras, o usuário percebe a grande facilidade de obter infinitos formatos com possibilidades de diagramação de suas idéias. Adaptado para o apoio ao ensino e à aprendizagem de crianças, muitos dos seus objetos, tal como o título do Mapa, tópicos principais, biblioteca, editores, notas, palavras-chave, entre outros, alteram os posicionamentos e conexões, com cores e atributos que são facilmente montados e interpretados.



Fig. 7.6. Tela inicial do Mind Mapper Jr.

# 8

## SIMULADOR ARTIFICIAL PARA AVALIAÇÃO DA APRENDIZAGEM DA LÓGICA

### 8.1. A identidade de um Software Educativo

Os critérios para uma avaliação de um software para uso educacional, segundo uma concepção construtivista de aprendizagem, deve refletir sobre a classificação dos mesmos quanto ao tipo e nível de aprendizagem proposta e depois medida.

A tarefa de avaliar o processo crítica e criteriosamente os resultados educacionais determina o uso dos computadores e de softwares no processo de aprendizagem, o que não dispensa a contextualização do conceito de avaliação do ponto de vista construtivista, promovendo a aprendizagem dos alunos e ajudando na construção do processo de conceituação e no desenvolvimento de habilidades importantes para participação direta do aprendiz na sociedade do conhecimento.

Numa perspectiva construtivista, a aprendizagem ocorre quando a informação é processada pelos esquemas mentais e agregada a esses esquemas. Assim, o conhecimento construído vai sendo incorporado aos esquemas mentais que são colocados para funcionar diante de situações desafiadoras e problematizadoras. Piaget aborda a inteligência como algo dinâmico, decorrente da construção de estruturas de conhecimento que, à medida que vão sendo construídas, vão se alojando no cérebro. A inteligência, portanto, não aumenta por acréscimo, e sim, por reorganização. Essa construção tem a base biológica, mas vai se dando à medida em que ocorre a interação, trocas recíprocas de ação com o objeto do conhecimento, onde a ação intelectual sobre esse objeto tenta retirar dele qualidades que a ação e a coordenação das ações do sujeito colocaram neles. O conhecimento lógico-matemático provém da abstração sobre a própria ação.

O desequilíbrio é fundamental para que haja a falha, a fim de que o sujeito sinta a necessidade de buscar o reequilíbrio, o que se dará a partir da ação intelectual desencadeada diante do obstáculo: A ABSTRAÇÃO REFLEXIVA. É na abstração reflexiva que se dá a construção do conhecimento lógico-matemático (inteligência), resultando num equilíbrio superior e na conseqüente satisfação da necessidade.

## **8.2. Aspectos pedagógicos do processo**

O software para ser educativo deve se constituir em um ambiente interativo que proporcione ao aprendiz investigar, levantar hipóteses, testá-las e refinar suas idéias iniciais, dessa forma o aprendiz estará construindo o seu próprio conhecimento.

O aprendiz lança mão de todas as estruturas de conhecimentos disponíveis, como: conceitos, linguagens e estratégias de aplicação para representar e explicitar os passos da resolução do problema.

A execução do processo devolve resultados fiéis e imediatos para o aprendiz. O resultado obtido é fruto somente do que foi solicitado. A reflexão sobre o que foi executado no computador, nos diversos níveis de abstração, pode provocar alterações na estrutura mental do aluno. O nível de abstração mais simples é a empírica, que permite a ação do aprendiz sob o objeto, extraíndo dele informações como cor, forma, textura, etc. A abstração pseudo-empírica permite ao aprendiz deduzir algum conhecimento da sua ação ou do objeto. A abstração reflexionante permite ao aprendiz pensar sobre suas próprias idéias. Esse processo de reflexão sobre o resultado do programa pode provocar o surgimento de uma das alternativas: Na primeira, a resolução do problema apresentado pelo computador corresponde às idéias iniciais do aprendiz e portanto não são necessárias modificações no procedimento e na Segunda, a necessidade de uma nova depuração do procedimento porque o resultado é diferente das idéias iniciais.

O processo de depuração dos conhecimentos acontece quando o aprendiz busca informações ou conceitos em outros locais e estas são assimiladas pela estrutura mental, passando a ser conhecimento e as utiliza no programa para modificar

ou não a descrição anteriormente definida. Nesse momento, repete-se o ciclo descrição - execução - reflexão - depuração - descrição. Levando em consideração esse ciclo, o software pode ser interpretado como a explicitação do raciocínio do aprendiz, fornecendo ingredientes importantes para o processo de construção do conhecimento.

Por se tratar de uma proposta Tutorial, caracteriza-se por transmitir informações pedagogicamente organizadas, como se fossem um livro animado, um vídeo interativo ou um professor eletrônico. A informação é apresentada ao aprendiz seguindo uma seqüência, e o aprendiz pode escolher a informação que desejar. A informação que está disponível para o aluno é definida e organizada previamente, assim o computador assume o papel de uma máquina de ensinar. A interação entre o aprendiz e o computador consiste na leitura da tela e acionamento do teclado ou mouse para estabelecimento da interação aprendiz-máquina. A sua limitação se encontra justamente em não possibilitar a verificação se a informação processada passou a ser conhecimento agregado aos esquemas mentais.

Quanto aos níveis de aprendizagem, a proposta pode apresentar três modelos de resultados, onde o aprendiz se posta de forma absolutamente passiva, devendo apenas memorizar e repetir passos anteriormente definidos. Porém, de forma relecional, o aprendiz pode também adquirir determinadas habilidades fazendo relações com outros conteúdos de outras fontes de informações. A grande validade porém, está na forma criativa e associativa de novos esquemas mentais, possibilitando a interação entre pessoas e tecnologias, compartilhando objetivos comuns e caracterizando-se como um aprendizado participativo.

### **8.3. Aspectos Técnicos do processo**

Ao ser analisado do ponto de vista técnico, uma vez que estes aspectos orientam para uma adequada utilização, o software deverá observar as características: mídias empregadas, qualidade de telas, interfaces disponíveis, clareza de instruções, compartilhamento em rede local e Internet, compatibilização com outros softwares,



hardware e funcionalidade em rede, importação e exportação de objetos, apresentação auto-executável, recursos hipertexto e hiperlink, disponibilidade de help-desk, manual técnico com linguagem apropriada ao instrutor, facilidade de instalação, desinstalação e manuseio, etc. Ainda do ponto de vista técnico, é sugerido um controle textual para avaliação do sistema proposto, onde a estrutura instrutor-aprendiz poderá, interativamente, avaliar não só os resultados dos casos práticos, como também a eficiência da ferramenta. Algumas de suas principais características são:

- Apresenta as instruções de forma clara
- Indica as possibilidades de uso
- Especifica os requisitos de hardware/software
- Apresenta a facilidade de instalação e desinstalação
- Fornece o manual de utilização com linguagem apropriada
- É compatível com outros softwares e hardware
- Funciona em rede
- Importa e exporta objetos
- É auto- executável?
- Possui recursos de hipertexto e hiperlink
- Dispõe de help-desk
- Apresenta facilidade de navegação

## **8.4. Processos metodológicos do SE proposto**

A propriedade mais importante da proposta é medir a habilidade de aprender dentro de um ambiente e, com isso, melhorar seu desempenho na absorção de novas informações e aquisição de conhecimentos, passando o indivíduo aprendiz a demonstrar novos comportamentos em função de uma transformação na sua condição mental. Isto pode ser feito através de um processo interativo de ajustes aplicados a suas características. O Simulador proposto interpreta o aprendizado quando o sistema

revela uma solução específica para cada classe de problemas, similares ou não, correlatos ou não.

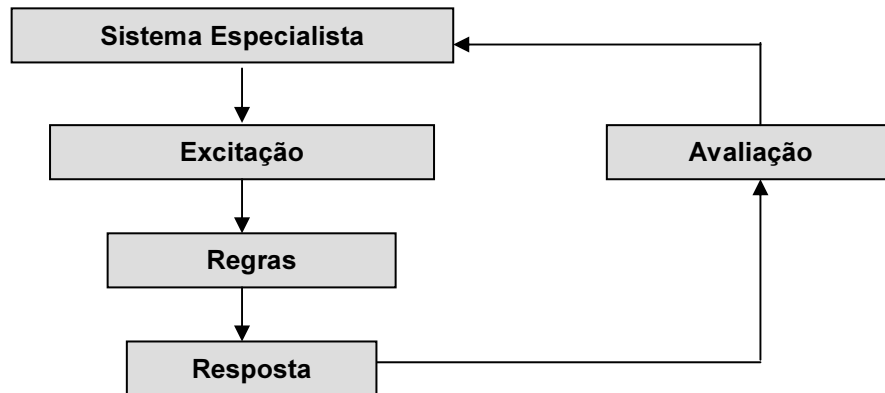


Fig. 8.1. Esquema metodológico do SE proposto

O SE, uma vez bem definido, apresentará uma base de conhecimentos contendo um conjunto de regras, que inclui o dispositivo de inferência, o administrador do conhecimento e as interfaces com o usuário, estabelecendo critérios geradores de explicações e justificativas para os dados conclusivos, promovendo desta forma, seus resultados, implicando diretamente na medição da aprendizagem da Lógica dos Teoremas principais da Álgebra de boole.

## 8.5. Descrição da ferramenta principal

A ferramenta utilizada nesta proposta é o Visual Manager. Este software adequado para organizar, classificar, recuperar e tratar informações em questões diretamente relacionadas a processos de aprendizagem. Apresenta-se como uma ferramenta moderna e poderosa, oferecendo ao usuário rapidez na elaboração de projetos especiais em operações extremamente elementares, com flexibilidade e significados lógicos obtidos durante todo o tempo em que processa as informações.



## MindManager 2002.Ink

Fig. 8.2. Ícone do MindManager na área de trabalho do Windows

Ao expandir as atividades, criando conceitos e adicionando sentenças na criação de “Branches” que se organizarão de forma lógica dentro do Mapa principal, o usuário se depara com facilidades para despertar a criatividade e o potencial de iniciativa para uma infinidade de idéias que serão direta ou indiretamente relacionadas.

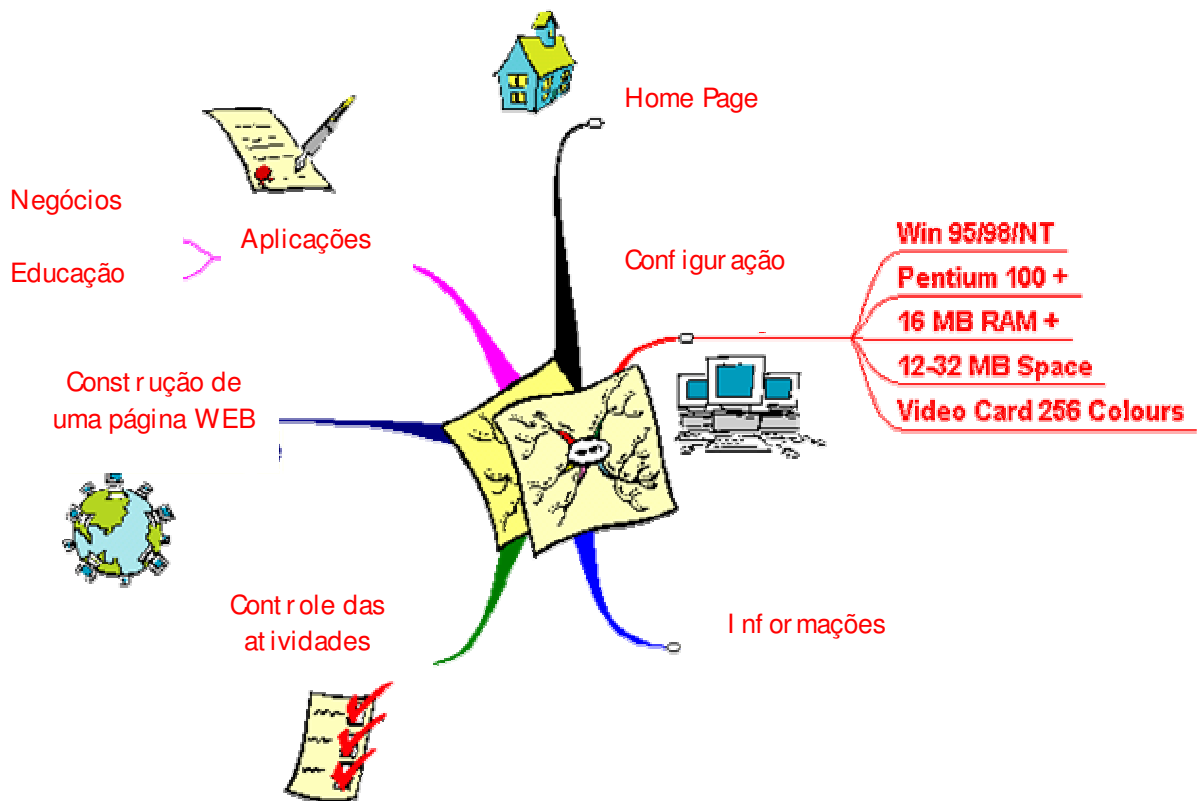


Fig. 8.3. Esquema funcional do Mind Manager

Outra grande vantagem do Mind Manager é a interação com o MS-Office, permitindo ao usuário a escolha do software ideal de acordo com sua necessidade mais imediata em qualquer ponto do mapa, com flexibilidade e controle integral sobre cada aplicação. As informações podem ser importadas ou exportadas pelos softwares do MS-Office em perfeita sincronia, uma vez definido o controle das atividades.

Durante o desenvolvimento de atividades criativas, os processos de apresentação podem assumir características de forma tal que se enquadram perfeitamente nas funções principais dos Softwares do MS-Office. Basta que no controle de atividades, o usuário gerente programe e interprete cada função do mapa, de acordo com o objetivo que se planeja alcançar.

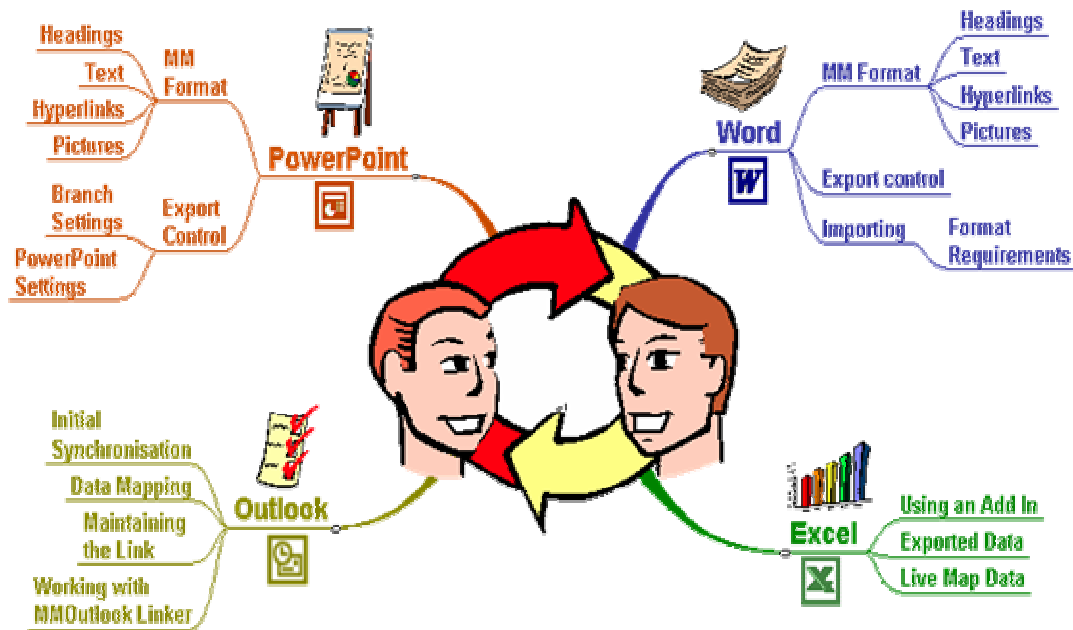


Fig. 8.4. Relações estabelecidas com o MS-Office.

## 8.6. Definições e procedimentos da aplicação

O SE proposto tem sua macroestrutura definida pelos passos de aplicação efetuados pelo aprendiz e as respostas características.

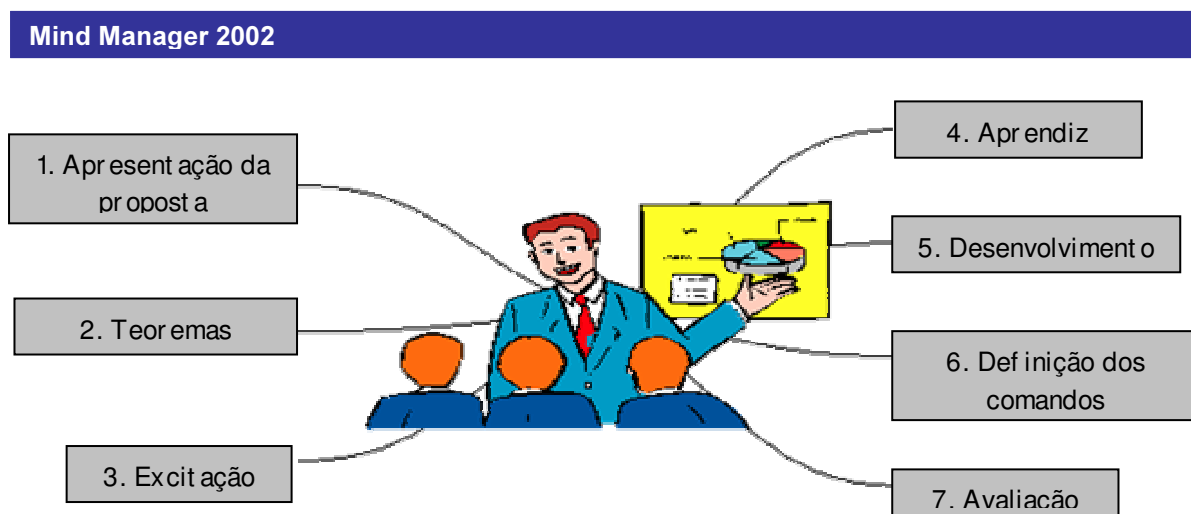


Fig. 8.5. Estrutura do trabalho de avaliação da aprendizagem da lógica

### 8.6.1. Tela de Apresentação: Proposta e Teoremas

O aprendiz terá a definição clara e objetiva dos dois Teoremas de Boole propostos para o desenvolvimento pelo Simulador.

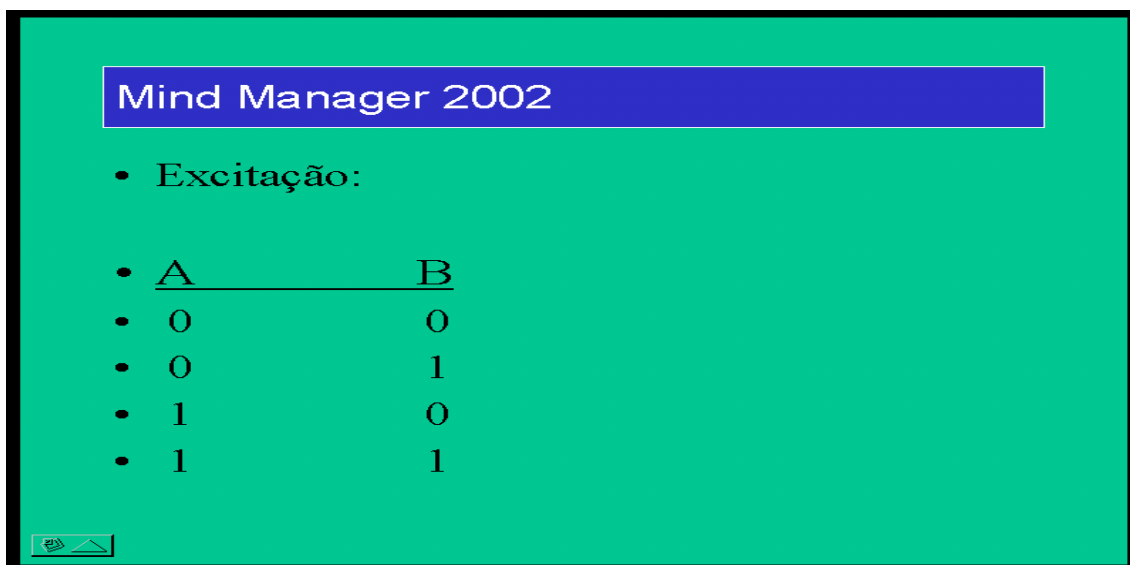
Mind Manager 2002

- **Proposta: Desenvolver, passo a passo, a comprovação de dois dos principais Teoremas da Álgebra de Boole.**
- **Teoremas**
  - **$(A \wedge B) \vee (A \wedge \neg B) \equiv A$**
  - **$(A \vee B) \wedge (A \vee \neg B) \equiv A$**

Fig. 8.6. Tela de apresentação do Sistema em Power Point

### 8.6.2. Excitação: Combinação das entradas

O sistema será excitado sempre com a combinação entre os operadores de entrada (A e B), de forma a produzir resultados sempre consonantes com o que estabelecem os Teoremas.



Mind Manager 2002

- Excitação:

<u>A</u>	<u>B</u>
• 0	0
• 0	1
• 1	0
• 1	1

Fig. 8.7. Tabela combinatória das Excitações do sistema em Power Point

### 8.6.3. A Intervenção do Aprendiz

O aprendiz, no momento em que acessar o Sistema simulador desenvolvido em Excel, terá à sua inteira disposição, as opções de entrada dos Operadores A e B, de forma tal que o próprio sistema se encarregará de simular todas as situações para que, finalmente, comprove se os Teoremas de Boole estão “Comprovados” ou “INCORRETOS”.

Inicialmente, é feita uma coleta das respostas do Aprendiz, de forma a ser armazenada na Planilha1 para depois ser comparada com os cálculos efetuados na Planilha2. O sistema de pontuação é muito simples, bastando a comparação entre as respostas do Aprendiz e os resultados dos cálculos efetuados de forma direta, o que incidirá as respostas “Ok”, para os acertos e “Equívoco” para as diferenças.

Nas células G4 e G5 serão digitados os operadores lógicos correspondentes a A e a B.

Sistema: Simulador da Aprendizagem da Lógica						
Teoremas da Álgebra de Boole						
Incremente o Operador A						1
Incremente o Operador B						1
A	B	$\sim B$	A e B	A ou B	A e $\sim B$	A ou $\sim B$
1	1	FALSO	VERDADEIRO	VERDADEIRO	FALSO	VERDADEIRO
Comprovação do 1º Teorema						
(A e B) ou (A e $\sim B$ )		Igual a A?	Aprendiz	Resultado		
VERDADEIRO		SIM	SIM	Ok!!!		
Comprovação do 2º Teorema						
(A ou B) e (A ou $\sim B$ )		Igual a A?	Aprendiz	Resultado		
VERDADEIRO		SIM	SIM	Ok!!!		

Fig. 8.8. Sistema simulador apresentado em Excel

#### 8.6.4. Definição dos comandos - Células

A8 = G4. Mera cópia do valor do operador, denominado A.

B8 = G5. Mera cópia do valor do operador, denominado B.

C8 = NÃO (B8). Inversor do operador B.

D8 = E(A8;B8). Correspondente a  $(A \wedge B)$ .

E8 = OU(A8;B8). Correspondente a  $(A \vee B)$ .

F8 = E(A8;NOT(B8)). Correspondente a  $(A \wedge \neg B)$ .

G8 = OU(A8;NOT(B8)). Correspondente a  $(A \vee \neg B)$ .

A12 = **OU(D8;F8)**. Corresponde a  $(A \wedge B) \vee (A \wedge \neg B)$ .

D12 = **SE(A12=FALSO;SE(A8=0;"SIM";"NÃO");SE(A8=1;"SIM";"NÃO"))**.

Corresponde a:

Se A12 = FALSO

Então

Se A8 = 0

Então "SIM"

Senão "NÃO"

Senão

Se A8 = 1

Então "SIM"

Senão "NÃO".

E12 = **SIM** ou **NÃO**. É a resposta do aprendiz.

F12 = **SE(D12=E12;"Ok!!!";"Equívoco")**.

Corresponde a:

Se D12 = E12

Então "Ok!!!"

Senão "Equívoco".

A16 = **OU(D8;F8)**. Corresponde a  $(A \wedge B) \vee (A \wedge \neg B)$ .

D16 = **SE(A12=FALSO;SE(A8=0;"SIM";"NÃO");SE(A8=1;"SIM";"NÃO"))**.

Corresponde a:

Se A16 = FALSO

Então

Se A8 = 0

Então "SIM"

Senão "NÃO"

Senão

Se A8 = 1

Então "SIM"

Senão "NÃO".



E16 = **SIM** ou **NÃO**. É a resposta do aprendiz.

F16 = **SE(D16=E16;"Ok!!!";"Equívoco")**.

Corresponde a:

Se D16 = E16

Então "Ok!!!"

Senão "Equívoco".

### **8.6.5. Avaliação dos resultados**

Os tempos de resposta podem se constituir em um aspecto determinante, uma vez que o aprendiz poderá ou não estar utilizando resposta compatível com a velocidade do processo cognitivo do ser humano.

É certo que os pré-requisitos de conhecimentos mínimos de lógica sejam exigidos para que se obtenha um mínimo de condições e credibilidade para os resultados, em especial, ao se considerar preceitos da psicologia no que se refere ao tempo de resposta, que é da ordem de dois segundos numa interlocução homem-máquina e à retenção dos dados na memória de curto termo, de dois a seis segundos, no máximo.

Tais resultados levam a concluir:

- perto de dois segundos: tempo de resposta ideal;
- de 2 a 4 segundos: impressão de espera, pode ser gerado pela memória de curto tempo;
- mais de 4 segundos: muito longo se o diálogo necessitar uma memorização a curto tempo e se não existem indicadores na tela do computador.

Em alguns casos, cabe distinguir algumas ocorrências que possam inviabilizar a precisão dos resultados de avaliação. Os erros de execução são facilmente detectáveis e retificáveis, provenientes do fato de se apertar inadvertidamente uma ou outra tecla que não a desejada e os erros de intenção que correspondem a uma má interpretação do sentido dos comandos ou mesmo, da significação dos procedimentos. Estes erros podem não ser detectados e sua retificação pode exigir um esforço de aprendizagem

por parte do aprendiz. Para a ocorrência de erros, há unanimidade sobre os elementos que devem ser assinalados imediatamente ou o mais rapidamente possível por causa da volatilidade da memória de curto tempo. Para a correção dessas ocorrências, o usuário pode rever facilmente a operação ou a linha onde se situa o erro e deve anular, eventualmente, na totalidade ou em parte o trabalho que foi feito.

Durante o processo é facultada ao aprendiz a possibilidade de transferir um dado ou instrução de uma aplicação para outra sem precisar reescrevê-los. Esta função permite minimizar os embargos e evitar os erros de digitação.

É utilizável um esquema na forma de Guia Funcional na forma de comandos para permitir ao aprendiz a todo e qualquer momento, uma explicação resumida sobre as propriedades dos Teoremas, dentro do estágio atual do trabalho, ou seja, uma explicação das funções e efeitos de um dado comando. Este tipo de possibilidade é clássica e possui grande popularidade.

<b>Tipo da questão</b>	<b>Forma canônica da questão</b>
Orientada a Objetivos	Que tipo de coisas eu posso fazer com este Programa?
Descritiva	O que é isso? O que é que isso faz?
Procedural	Como é que eu faço isto?
Interpretativa	Como foi que isto aconteceu?

## CONCLUSÃO

Apesar de existirem várias pesquisas sobre avaliação de software educacionais, a maioria delas preocupa-se apenas ou com aspectos de *hardware* ou com a quantidade de situações de aprendizagem. Esta proposta, entretanto, se foca na análise qualitativa de ambientes computacionais e o seu interesse em particular é desenvolver e despertar uma metodologia de avaliação de forma interativa.

É mostrado neste trabalho que as demonstrações de que a seleção de métodos e técnicas de treinamento é de extrema importância para medição de desempenho de aprendizagem e podem ser realizados com segurança e atingir os objetivos propostos em seu planejamento a partir de sistemas virtuais de controle.

A seleção das técnicas é fundamental na montagem de um programa de avaliação de aprendizagem e uma técnica adequada e bem posicionada em relação aos aprendizes fará com que o aprendizado necessário seja facilmente alcançado.

Houve uma preocupação inicial de se definirem telas bem diagramadas, sem necessidade de utilização de recursos sonoros nem de animação, pela própria natureza da proposta.

A proposta deste trabalho impõe um regime de interatividade, atratividade, intuitividade e uma necessidade despertada no aprendiz ao ter contato direto com o sistema, o que por motivação provoca e induz ao aprendizado de uma lógica não tão simples de ser compreendida.

Apesar deste trabalho ter sido concluído como se previu inicialmente, muito se há por fazer para que se atinjam os objetivos propostos, na forma de desempenho, representação e modelagem, armazenamento de dados e resultados para futuras comparações para fins de avaliação de evolução, comprovação de muitas outras teorias e teoremas correlatos, entre outras possíveis falhas de qualquer natureza.

Definir, projetar e implementar um sistema de apoio ao processo ensino-aprendizagem envolve questões de ordem tecnológica, humana, organizacional, econômica e legal, o que pode implicar na exigência de uma estrutura não tão simples.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ACKLEY**, D. H. & **LITTMAN** M. L., Interactions between Learning and Evolution, 1992, pp. 487-509,
- ALBERDI**, E. & **MATUTE**, H., Aprendizaje a partir de ejemplos y asignacion de pesos en nuevas tendencias en Inteligencia Artificial, Rementeria, Santiago, 1992.
- ALMEIDA**, M. A. F., Aprender, atividade inteligente: e se esta ingeligência for parcialmente artificial?, Mestrado em Ciências da Computação: Universidade Federal de Santa Catarina, Florianópolis, 1999.
- ANDERSON**, J. R., Adaptive character of thought, Lawrence Erlbaum Associates, 1990.
- BARRETO**, J. M. & **NEYER** & M. de & **LEFÈVRE**, Jacques & **GOREZ**, Reymond., Qualitative Physics versus Fuzzy Sets Theory in Modeling and Control. In: IECON'91 IEEE International conference on industrial electronics, control and instrumentation, Kobe, 1991, pp. 1651-1656.
- BARRETO**, J. M. & **NEYER**, M. Qualitative and Quantitative Models of Systems. In: **HANUS**, R. & **KOOL**, P. & **TZAFESTAS**, S. (Org.). Mathematical and Intelligent Models in System Simulation. Amesterdam, 1991, pp. 269-274.
- BARRETO**, J. M. e **CALOBA**, L. , III Congresso brasileiro de redes neurais, 1997.
- BARRETO**, J. M., "Inteligência Artificial no limiar do Século XXI", 2ª Ed. Revista e aumentada, Duplic. Florianópolis-SC, 2000.
- BARRETO**, J. M.; A Topological Model for Qualitative Time. In: AINN-90: Artificial intelligence applications and neural networks, 1990, Zurich.
- BARRETO**, J. M.; **PAGANO**, Rosane Lessa. The Role of Qualitative Simulation in Computer Assisted Learning. In: ICTE'91 International conference in technology and education, 1991, Toronto, Ontario. 1991. pp. 569-671.
- BEER**, R. D., Intelligence as adaptive behavior, Academic Press, 1990.

**BESSIERE**, P., "Genetic Algorithms Applied to Formal Neural Networks: Parallel Genetic Implementation of a Boltzmann Machine and Associated Robotic Experimentations", 1992, p.310-314.

**BESSIERE**, P., Genetic algorithms applied to formal neural networks: parallel genetic implementation of a Boltzmann machine and associated robotic experimentations, 1991, P.310-314,

**BOOG**, G. G. "Manual de Treinamento, J. R. "AI in CAI: Na artificial intelligence approach to computer assisted instruction". IEEE Transactions on Man-machine system, Vol. I, N. 4, pp. 190-202, 1970.

**BOSCOLO**, H., Casos na informatização, Jornal O Estado de São Paulo, 1998.

**BOURGINE**, P. & VARELA, F. J., "Introduction: Towards a Practice of Autonomous Systems, 1991, p.xi-xvii,

**BROWN**, J. S. & BURTON, R. R. & Bell, A. G., "Sophie: A step toward a reactive learning environment", International journal of Man-machine studies, Vol I, pp. 675-696, 1975.

**BURTON**, R. R. e Brown, J. S., "An investigation of computer coaching for informal learning activities", International journal of Man-machine studies, Vol. 11, pp. 5-24, 1979.

**CAMPBELL-KELLY**, M., "Charles Babbage: passages from the life of a phylosopher", 1994, Piscataway, New Jersey, IEEE Press.

**CARBONELL**, J. R. "AI in CAI: Na artificial intelligence approach to computer assisted instruction". IEEE Transactions on Man-machine system, Vol. I, N. 4, pp. 190-202, 1970.

**CARVALHO**, G. M. R. e TAVARES, Márcia da Silva. "Informação & Conhecimento", Rio de Janeiro-RJ, Qualitymark Ed., 2001.

**CLANCEY**, W. J. , "Knowledge-based tutoring: The Guidon program", The MIT Press, 1987a.

**CLANCEY**, W. J. , "Methodology for building na intelligent tutoring system", The MIT Press, 1987b.

**CLÁUDIO**, D. M., DIVERIO, T. A . e TOSCANI, L. V., "Fundamentos da matemática computacional", Sagra, Porto Alegre-RS, 1987.

- DICKERSON**, R. E. Dickerson, "Chemical Evolution and the Origin of Life", 1978, pp. 30-47, Evolution, San Francisco, Freeman and Co.
- DRESCHER**, G. L., Made up minds, MIT Press, 1991.
- DRESS**, W. B., "Electronic Life and Synthetic Intelligent Systems", 1990, Instrumentation and Controls Division, Oak Ridge National Laboratory.
- DRESS**, W. B., "In-Silico Gene Expression: A Specific Example and Possible Generalizations", 1990, Proceedings of Emergence and Evolution of Life-Forms.
- DUBOIS**, D. e **PRADE H.**, *"Fuzzy Sets and Systems: Theory and Applications"*, Academic Press, 1980.
- FAUSETT**, L. "Fundamental of neural networks: Architectures, algorithms and applications. Prentice Hall international Inc., New Jersey, 1994.
- FAYYAD**, U. M., **PIATETSKY-SHAPIRO**, G. e **SMITH**, P. "From data mining to knowledge discovery: na overview", Vol 1 of Fayyad et al., Cap. 1 pp. 1-34, 1996.
- FISHER**, D. & **LANGLEY**, P. The structure and formation of categories en The psychology of learning and motivation, 1994, Ed. Bower, G. H., Academic Press.
- FULLMER B.** & **MIKKULAINEN R.**, "Using Marker-Based Genetic Encoding of Neural Networks to Evolve Finite-State Behaviour", 1992,
- GARIS**, H. de, "Steerable {G}en{N}ets: the genetic programming of steerable behaviors in {G}en{N}ets", 1992, p.272-281.
- GOLEMAN**, D. P., "Emotional Intelligence: Why it can matter more than {IQ} for character, health and lifelong achievement", 1995, USA, Bantam Books.
- GOREZ**, Reymond; **NEYER**, Marc de; **GALARDINI**, Daniele; **BARRETO**, J. M., Model based control systems: Fuzzy and qualitative realizations. In: SICICA/92 IFAC Symposium on intelligent components and instruments for control applications, 1992, Malaga, 1992. Pp. 681-686.
- GUZMÁN**, M. de, Aventuras matemáticas. Una ventana hacia el caos y otros episodios Pirámide, Madrid, 1995.
- GUZMÁN**, M. de, Para pensar mejor. Desarrollo de la creatividad a través de los procesos matemáticos, Pirámide, Madrid, 1994.

**GRATTO**, Karen Smith - Toward Combining Programmed Instruction and constructivism for Tutorial Design.

[http://www.coe.uh.edu/insite/elec\\_pub/html1995/199.htm](http://www.coe.uh.edu/insite/elec_pub/html1995/199.htm)

**HADAMARD**, J., Matemáticas en el mundo moderno, pp. 14-17, Madrid, Blume, 1974, Nueva York, Princeton Univ. Press, 1954.

**HANNA**, G., Towards Gender Equity in Mathematics Education. An ICMI Study (Kluwer, Dordrecht, 1996)

**HARVEY**, I. "Species Adaptation Genetic Algorithms: a basis for a continuing", 1991, p. 346-354.

**HENDERSON** B. & **JOHNSON** J. E., "The object-oriented systems life cycle, Journal communications of the ACM, 1990, Vol.33, N°9, p.142—159.

**HINTON**, G. E., Parallel models of associative memory, Lawrence Erlbaum Associates, 1989.

**HOFSTADTER**, D. R. & **GÖDEL**, E. Un eterno y gracil bucle, Tusquets Editores & CONACYT, 1987.

**HOLTON**, D. (editor), The Teaching and Learning of Mathematics at University Level. An ICMI Study, Kluwer, Dordrecht, 2001.

**JEFFERSON** D. & others, "Evolution as a theme in artificial life: The genesys/tracker system", 1992, pp. 549-578.

**JEFFERSON**, D. & **COLLINS**, R. & **COOPER**, C. & **DYER**, M. & **FLOWERS**, M. & **KORF**, R. & **TAYLOR**, C. & **WANG**, A., "Evolution as a Theme in Artificial Life: the Genesys/Tracker System", 1990, UCLA-AI-90-09, Computer Science, UCLA.

**KING**, P. & Mamdani, E., The application of Fuzzy control systems to industrial processes. Automatica 13 1977, pp.235-242.

**KLAHR**, D.(Ed.), Production system models of learning and development, MIT Press, 1987.

**KLEIN**, F., Matemática elemental desde un punto de vista superior. Aritmética - Álgebra Análisis, Biblioteca Matemática, Madrid, 1927.

**KLEIN**, S. B., Aprendizaje, principios y aplicaciones, McGraw Hill, 1994.

**KODRATOFF**, Y. Introduction to machine learning, Morgan Kaufmann Publishers, 1988.

- KOZA**, J. R. "Evolution of subsumption using genetic programming, 1992, pp.110-119.
- LAIRD**, J. & **ROSENBLOOM**, P. S. Universal subgoaling and chunking Kluwer Academic Publishers, 1986.
- LANGTON** C. & **TAYLOR**, C. & **FARMER** J. and **RASMUSSEN** S., "Artificial Life II", Reading, Massachusetts, Addison-Wesley.
- LANGTON** C. and others, "Artificial Life II", 1991, Reading, Massachusetts, Addison-Wesley.
- LANGTON**, C. G. & **TAYLOR**, C. & **FARMER**, Doyne & **RASMUSSEN**, Steen, Artificial Life II: Proceedings of the Workshop on Artificial Life}, 1992, Redwood City, CA.
- LANGTON**, C. G. (Ed.), 1989 Artificial life - The proceedings of an interdisciplinary workshop on the synthesis and simulation of living systems, Addison-Wesley.
- LANGTON**, C. G., Artificial life II - Proceedings of the workshop on artificial life Addison-Wesley, 1990.
- LANGTON**, C. G., "Artificial life: an overview", 1995, Cambridge, Massachusetts, MIT Press.
- LANGTON**, C. G., Artificial Life III: Proceedings of the Workshop on Artificial Life}, 1994, Reading, MA, 1992.
- LANGTON**, C. G., Artificial Life: the Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems}, 1989, Redwood City, CA.
- LIPPMANN**, R. P. "A n introduction to computing with neural networks". IEEE Acoustics, Speech and signal processing (ASSP) Magazine, V. 4, n.2, pp. 4-22, 1987.
- MATUTE**, H., Aprendizaje y representacion de conceptos en aprendizaje y memoria Humana: Aspectos Basicos Y Evolutivos, J.I. Navarro Ed., 1993, McGraw Hill.
- MENCZER**, F. & **PARISI**, D. A model for eht emergence of Sex in evolving networks: Adptive Advantage or Random Drift?, 1992, pp. 337-345,
- MENCZEWR**, F. & **BELEW**, R. K., Evolving Sensors in Environments of Controlled Complexity, 1994, Brooks, R. and Maes, P., Artificial Life IV, Cambridge, MA, MIT Press.
- MIKAMI**, S. & **TANO**, H. "& **KAKAZU**, Y., An Autonomous Legged Robot That Learns to Walk Through Simulated Evolution, 1993, pp.758-767, Self-Organisation and Life, From



Simple Rules to Global Complexity, Proceedings of the Second European Conference on Artificial Life, MIT Press, Cambridge.

**MITCHELL M. & FORREST S. & HOLLAND J. H.**, The royal road for genetic algorithms: fitness landscapes and (GA) performance, 1991, pp. 245-254.

**MITCHELL, M. and FORREST, S.**, Genetic Algorithms and Artificial Life, Artificial Life, 1993.

**MONTEIRO, R. D. V.**, Simulador em Excel. Esta dissertação apresentada para o Programa de Mestrado em Ciência da Computação, Florianópolis-SC, 2002.

**MORAIS, F.**, Uma abordagem cognitiva das organizações. Estudos de mapeamento cognitivo na banca portuguesa. Porto: FPCE-UP.

**MOREIRA, M. A .**, “Teorias de aprendizagem”, EPU, São Paulo-SP, 1999.

**MORRIS, H. C.**, On the feasibility of computational artificial life en from animals to animats, 1991, Meyer, Jean-Arcady y Wilson, Stewart W. Ed., MIT Press.

**NASCIMENTO, C. L. Jr. e YONEYAMA, T.**, “Inteligência artificial em controle e automação”, Edgard Blucher Ltda., 2000.

**NEYER, M. de & GOREZ, Reymond & BARRETO, J. M.**, Comparative analysis of control system with systems with Fuzzy or Qualitative Internal Models. In: ESC'91 European simulation conference: Intelligent process control and scheduling, Het Pand, Ghent, Bélgica: Society for Computer Simulation, 1991, pp. 145-150.

**OHLSSON, S.**, The interaction between knowledge and practice in the acquisition of cognitive skills en foundations of knowledge acquisition, Susan Chipman (Ed.), Kluwer Academic Publishers.

**PAGANO, Rosane L. & BARRETO, J. M.**, “Laboratory experiment simulation in electrical engineering education”, Proc. of ESC'89 – 3<sup>rd</sup> European Simulation Congress, Edinburgh, 1989.

**PAGANO, Rosane L. & BARRETO, J. M.**, “Psychodrame pédagogique et hypermédia”, Proc. of Sixième Congrès APTLF, Brussels, 1990.

**PARISI, D. and NOLFI, S. and CECCONI, F.**, Learning, Behaviour and Evolution, 1991.

**PEARCE, J. M.** A model for pavlovian conditioning, Psychological Review, 1980.

**PEDRYCZ, W.** “Fuzzy control and Fuzzy systems”, John Wiley, 1993.

**PELLEGRINI**, G. F., **COLLAZOS**, K. & **BARRETO**, J. M., “Extração de conhecimento a partir dos sistemas de informação”, Florianópolis, 2000.

**PINHO**, A . A . “Introdução à Inteligência Artificial”, Rio de Janeiro, 1999.

**PIROLI**, P., Towards A unified model of learning to program en cognitive models And intelligent environments for learning programming, Lemut, Enrica (Ed.), Nato ASI Series. Series F, Computer and Systems Sciences, Vol. 111.

**RESCORLA**, R. A. & **WAGNER**, A. R., A Theory of Pavlovian conditioning en classical conditioning II, Black, A. H. Ed., Appleton-Century-Crofts.

**RODRIGUEZ**, M. & **ERARD**, P., **MULLER**, J. P. Ed., Virtual Environments For Simulating Artificial Autonomy En Artificial Life And Virtual Reality, 1994, Magnenat Thalmann, Nadia (Ed.) John Wiley & Sons.

**ROSENBLOOM**, P. S. Soar papers: Research on integrated intelligence, volúmenes 1y2.

**SABBATINI**, Renato M. E. “A Informática Ubíqua”. Campinas-SP, Jornal Correio Popular, 1996.

**SANDRI**, S. A ., “Introdução a Lógica Fuzzy”, Inpe, 1996.

**SCHLAGER**, K. J., On-line fluorometric microbiological analysis for life support systems, 1992, pp.1-10, SAE.

**Scientific American Selection**, Life and death and medicine, 1973, pp.147, San Francisco, Freeman, Readings from Scientific American.

**SEABRA**, Carlos - Software educacional e telemática: novos recursos na escola  
<http://penta2.ufgrs.br/edu/edu3375/leciona.html>

**SILVA** , Dirceu - Informática e Ensino: visão crítica dos softwares educativos e discussão sobre as bases pedagógicas adequadas ao seu desenvolvimento - -  
<http://penta2.ufgrs.br/edu/edu3375/leciona.html>

**SMITH**, E.E., Foundations of cognitive science, 1989, MIT Press.

**STADDON**, J. E. R., Adaptive behavior and learning, Cambridge University Press, 1983.

**STRUDWICK**, Janete - Behaviourist and Constructivist approaches to multimedia -  
<http://penta2.ufgrs.br/edu/educ3375/e3375.html>

**TANSCHHEIT**, R. & **BARRETO**, J. M., Controle de misturador de fluídos na matemática nebulosa. II Congresso brasileiro de Automática, Florianópolis-SC, 1978.

**TANSCHHEIT**, R. & **BARRETO**, J. M., Controle de processo por Lógica Nebulosa. 1978, Dissertação de Mestrado em Engenharia Elétrica, Instituto Militar de Engenharia.

**TANSCHHEIT**, R. & **BARRETO**, J. M., Controle de processos por matemática nebulosa: Resultados experimentais. In: XXXI Reunião Anual da SBPC, Fortaleza-CE, 1979.

**TERADA**, R., "Desenvolvimento de algoritmos e estruturas de dados", Makron Books, São Paulo-SP, 1991.

**VAARIO** J., Artificial Life as Constructivist (AI), Journal of SICE (Japanese Society of Instrument and Control Engineers), 1994, Vol. 33, nº1, pp.65-71.

**VALENTE**, J. Armando - Análise dos diferentes tipos de Softwares usados na Educação - NIED - UNICAMP - E-mail: [jvalente@turing.unicamp.br](mailto:jvalente@turing.unicamp.br)

**VARELA** & Bourguine, Proceedings of the First European Conference on Artificial Life. Toward a Practice of Autonomous Systems, 11-13 Dez, Paris, France, MIT Press.

**VARELA** F. J. & Bourguine P., "Proceedings of the First European Conference on Artificial Life. Towards a Practice of Autonomous Systems", Paris, MIT Press, Cambridge, MA.

**VARELA**, F. J. & and **BOURGINE** P., "Proceedings of the First European Conference on Artificial Life. Towards a Practice of Autonomous Systems", Paris, MIT Press, Cambridge, MA.

**VARELA**, F. J. & **BOURGINE** P., "Proceedings of the First European Conference on Artificial Life. Toward a Practice of Autonomous Systems, Paris, France, MIT Press, Cambridge, MA.

**VARELA**, F. J. & **BOURGINE** P., "Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life", Cambridge, MA, USA, MIT Press.

**VARELA**, F. J. & Bourguine P., First European Conference on Artificial Life. Toward a Practice of Autonomous Systems, 1991, Paris, Press.

**VARELA**, F. J. & **BOURGINE**, P., Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life, Cambridge, MA, USA, MIT Press.

**VARELA**, F. J. & **BOURGINE**, P., "First European Conference on Artificial Life", Paris, The MIT Press.

**VARELA**, F.J. & **BOURGINE**, P., "Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life", Cambridge, MA, USA, MIT Press.

**VARELA**, F.J. & **BOURGINE**, P., "Proceedings of the First European Conference on Artificial Life. Toward a Practice of Autonomous Systems", Paris, MIT Press, Cambridge, MA.

**VENTURINI**, G., Characterizing the adaptation abilities of a class of genetic based machine learning algorithms, 1991, pp.302-309, Ed. F. J. Varela and P. Bourgine, Proceedings of the First European Conference on Artificial Life. Toward a Practice of Autonomous Systems, 11-13 Dez, Paris, França, MIT Press, Cambridge, MA.

**WEBB**, B. & **Smithers**, T., The Connection between (AI) and Biology in the Study of Behavior, 1991, pp.421-428, Ed. Francisco J. Varela and Paul Bourgine, "First European Conference on Artificial Life", Dez, "A Bradford Book, Paris, The MIT Press.

**WERNER** G. M. & **Dyer** M. G., Evolution of Communication in Artificial Organisms, 1991, pp.659-687, Artificial Life II: Proceedings of the Workshop on Artificial Life Held in 1990.

**WILLIAMS**, B.V. & **Bounds**, D.G., Learning and Evolution in Populations of Backprop Networks, 1993, Proceedings of ECAL93 - European Conference on Artificial Life.

**YAEGER**, Larry, Computational Genetics, Physiology, Metabolism, Neural Systems, Learning, Vision and Behavior or PolyWorld: Life in a New Context, 1994, pp.263-298, Ed. C. G. Langton, "Artificial Life III, Proceedings Volume XVII", Addison-Wesley, Org. Santa fe institute studies in the sciences of complexity.