



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

Maria Aparecida Fernandes Almeida, M. Sc.

**HIPERTÔMATOS NA COMPUTAÇÃO
APLICADA À EDUCAÇÃO**

Tese submetida à Universidade Federal de Santa Catarina como parte dos requisitos para obtenção do grau de Doutor em Ciência da Computação.

Jorge Muniz Barreto, D. Sc.A.

Orientador

Florianópolis, 22 de Fevereiro de 2002.

HIPERTÔMATOS NA COMPUTAÇÃO APLICADA À EDUCAÇÃO

Maria Aparecida Fernandes Almeida

Maria Aparecida Fernandes Almeida, M. Sc.

Esta Tese foi julgada adequada para a obtenção do título de Doutor em Ciência da Computação, área de concentração - Sistemas de Conhecimento, e aprovada em sua forma final pelo Programa de Pós-graduação em Ciência da Computação da Universidade Federal de Santa Catarina.

[Signature]

Prof. Fernando A. O. Gauthier, Dr.
Coordenador do Programa

Banca Examinadora:

[Signature]

Prof. Jorge Muniz Barreto, D.Sc.A, INE-UFSC

[Signature]

Prof. Raul César Batista Martins, Dr., PUC-RJ

[Signature]

Prof.^ª. Maria Augusta Soares Machado, Dra., IBMEC-RJ

[Signature]

Prof.^ª. Aurora Trinidad Ramírez Pozo, Dra., UFPR

[Signature]

Prof. Mauro Roisemberg, Dr., INE-UFSC

[Signature]

Prof. Paulo Sérgio da Silva Borges, Dr., INE-UFSC

Florianópolis, 22 de fevereiro de 2002.

*“Jorge sentou praça
Na cavalaria
E eu estou feliz porque eu também
Sou da sua companhia
Eu estou vestida com as roupas
E as armas de Jorge
Para que meus inimigos tenham mãos
E não me toquem
Para que meus inimigos tenham pés
E não me alcancem
Para que meus inimigos tenham olhos
E não me vejam
E nem mesmo um pensamento
Eles possam ter para me fazerem mal
Porque eu estou vestida com as roupas
E as armas de Jorge
Salve Jorge
Salve Jorge
Salve Jorge
Armas de fogo
O meu corpo não alcançarão
Facas e espadas se quebrem
Sem o meu corpo tocar
Cordas e correntes arrebentem
Sem o meu corpo amarrar
Porque eu estou vestida com as roupas
E as armas de Jorge”*

“Jorge de Capadócia”, Jorge Ben.

Dedico esta tese à Jorge Muniz Barreto.

Agradecimentos

Agradeço a *todas* as pessoas que colaboraram, ao longo destes anos, direta ou indiretamente, na realização deste trabalho.

Os meus agradecimentos especiais vão para o meu orientador, Professor Jorge Muniz Barreto, pessoa a quem dedico esta tese, pelo que aprendi sob sua orientação e pela sua atuação na implantação do Doutorado em Ciência da Computação da UFSC.

Agradeço aos membros das bancas examinadoras (exame de qualificação e tese) que contribuíram, de forma tão valiosa, nas sugestões e na avaliação deste trabalho.

Às instituições envolvidas por todo apoio recebido.

Aos alunos, que foram grandes colaboradores e incentivadores da pesquisa desenvolvida.

Aos colegas, amigos, parentes ...

À DEUS!

Sumário

Sumário	i
Lista de Figuras	v
Lista de Tabelas	vii
Lista de Abreviaturas	viii
Lista de Símbolos	xi
Lista de Publicações	xii
Resumo	xv
Abstract	xvi
1 Introdução	1
1.1 Motivação	1
1.2 Objetivos	5
1.2.1 Objetivo Geral	5
1.2.2 Objetivos Específicos	6
1.3 Estrutura do trabalho	6
2 Máquinas para Ensinar...	8
2.1 Máquina de Pressey	10
2.2 Máquinas para Ensino: fonógrafo e televisão	10
2.3 Máquinas de Skinner	11
2.4 Máquina de Crowder	14

2.5	Máquina de Pask	14
2.6	Máquina para Ensinar: O computador	16
2.7	Inteligência Artificial no Ensino	24
2.8	Aprendendo no Ciberspaço	28
2.9	Considerações	31
3	Cognição e Computação	34
3.1	Ciência Cognitiva	34
3.2	Psicologia e Filosofia Computacional	37
3.3	Cognição e Computação	38
3.4	Paradigmas da Ciência Cognitiva	39
3.4.1	Ciência Cognitiva Computacional Simbólica	39
3.4.2	Ciência Cognitiva Computacional Conexionista	40
3.5	Aprendizado Animal × Aprendizado de Máquina	41
3.6	Taxinomia do Aprendizado de Máquina	43
3.6.1	Estratégias de aprendizado	43
3.6.2	Representação do conhecimento	47
3.6.3	Domínio de aplicação	50
3.6.4	Classificação segundo a retroação do mundo	50
3.6.5	Classificação quanto a finalidade do aprendizado	51
3.6.6	Classificação quanto a interação com o meio	52
3.6.7	Classificação quanto ao tipo de conhecimento adquirido	52
3.7	Aprendizado: faceta importante da Inteligência	53
3.7.1	O Especialista e o Generalista	53
3.7.2	O problema da medida métrica da inteligência	54
3.7.3	A Inteligência Humana é uma Função Computável?	56
4	Sistemas Formais e Teoria das Categorias	60
4.1	O que são sistemas formais?	60
4.2	Origens dos sistemas formais	61
4.3	Métodos Formais, Engenharia de Programas e Categorias	62
4.4	Representação de Sistemas Formais	64
4.5	Linguagens e Gramáticas Formais	68

4.6	Teoria das Categorias	70
4.6.1	Teoria das Categorias: definições	71
4.6.2	Um Sistema Formal gera uma Categoria	71
4.6.3	Representação de um Morfismo	72
4.6.4	Diagramas	72
4.6.5	Tipos de morfismos	72
4.6.6	Dualidade	74
4.6.7	Funtores	74
4.6.8	Categoria dos Conjuntos	75
4.6.9	Categoria de Conceitos	77
4.6.10	O Problema do Contexto na Categoria de Conceitos	78
4.7	Outras Categorias	81
5	Hipertômatos	83
5.1	Definição teórica de Hipertexto	83
5.2	Conceitos básicos da Teoria dos Autômatos	84
5.3	Hipertexto como autômato	88
5.4	Estados observáveis e estados alcançáveis	91
5.5	Definição de Hipertômato	92
5.6	Categoria dos Autômatos	94
5.7	Categoria de Estados de um Hipertômato	94
5.8	Categoria dos Hipertômatos	95
5.9	Modelagem categórica: um novo caminho para um velho problema	97
5.10	Nós essenciais de um Hipertômato	99
5.11	Metodologia para Construção de Hipertômatos	101
5.12	Hipertexto x Ensino de RNA: a inspiração	103
5.13	Exemplo 1: Um ambiente de Ensino de Computação	106
5.14	Exemplo 2: Hipertômato para Ensino de Neurofisiologia	109
6	Hipertômato e Ciclo de Vida	112
6.1	Ciclo de Vida em Programas Educacionais	112
6.2	Resultados de uma pesquisa geral	114
6.3	Pesquisa com usuários de um ambiente para Ensino de ICC	127

<i>SUMÁRIO</i>	iv
7 Epílogo	140
Referências Bibliográficas	143
Índice Remissivo	162

Lista de Figuras

2.1	B. F. Skinner . Fonte: UPI- The Bettmann Archive	11
2.2	Discoverer: programas lineares. Fonte: [44]	12
2.3	Máquina Devereux na França. Fonte: [45]	12
2.4	Auto-Tutor (EUA)	13
2.5	Máquina Didak Fonte: [221]	15
2.6	Janela da Máquina Didak Fonte: [221]	16
2.7	EnBasic: adaptação dos programas do PLATO. Fonte: [45]	21
2.8	TUTOR: versões para computadores Apple II, Commodore 64 e PC .	22
2.9	Entradas e Saídas Motoras mediadas pelo computador[81]	28
2.10	“N.A.C. Eye”[81]	29
3.1	Taxinomia do Aprendizado segundo Carbonell [52]	43
4.1	Sistema Formal Gerando Categoria	72
4.2	Representação dos objetos e morfismos numa Categoria	72
4.3	Monomorfismo	73
4.4	Epimorfismo	73
4.5	Isomorfismo	73
5.1	Grafo representando o autômato M	88
5.2	Mudança de saída e inalteração de estado no hipertexto H_p	91
5.3	Representação de um hipertexto H	92
5.4	Evolução dos modelos de Pagano e Almeida	93
5.5	Modelo generalizado: Categoria de Hipertômatos	95
5.6	Categoria de Estados de um Hipertômato	96
5.7	Categoria de Hipertômatos	96

5.8	Functor Isomórfico associando Hipertômatos	97
5.9	Exemplo de não monomorfismo na navegação	98
5.10	Nós essenciais de um Hipertômato	99
5.11	Nós essenciais do portal do Projeto ICEM	102
5.12	Página inicial do portal do Projeto ICEM	102
5.13	Tela de um hipertexto simples	103
5.14	Grafo de um Hipertômato para Ensino de Redes Neurais	105
5.15	Tela de apresentação do ambiente	107
5.16	Tópico de uma aula expositiva disponível para acesso	108
5.17	Acompanhamento de uma tarefa dada	109
5.18	Grafo de um Hipertômato para Ensino de Computação	110
5.19	Grafo de um protótipo para Ensino de Neurofisiologia [214]	111
6.1	Principais usos do computador	118
6.2	Livro ou hipertexto?	122
6.3	Ensino menos fatigante?	124
6.4	Ler na tela ou imprimir?	125

Lista de Tabelas

2.1	Alguns ITS históricos - adaptado de [120]	26
3.1	Aprendizado Animal \times Aprendizado de Máquina	42
5.1	Transição de estados do autômato M	88
6.1	Resultados dos Opinários 1 e 2	115
6.2	Resultado em cada curso	117
6.3	Resultados no CCA	139

Lista de Abreviaturas

1. CAI - Computer-Aided Instruction.
2. CAL - Computer-Aided Learning.
3. CCA - Centro de Ciências Agrárias.
4. CEE - Comissão Económica Europeia.
5. CMI - Computer Managed Instruction.
6. GPEB - Grupo de Pesquisas em Engenharia Biomédica.
7. GUI - Graphical User Interface.
8. HTML - Hypertext Markup Language.
9. ICEM - Inteligência Computacional no Ensino com Multimídia.
10. III CBRN - III Congresso Brasileiro de Redes Neurais.
11. IEEE - The Institute of Electrical and Electronics Engineers.
12. IA - Inteligência Artificial.
13. IAC - Inteligência Artificial Conexionista.
14. IAS - Inteligência Artificial Simbólica.
15. IBM - International Business Machines.
16. ICAI - Intelligent Computer-Aided Instruction.

17. ICAL - Intelligent Computer-Aided Learning.
18. ICC - Introdução à Ciência da Computação.
19. ICEM - Inteligência Computacional no Ensino com Multimídia.
20. INE - Departamento de Informática e de Estatística.
21. ITS - Intelligent Tutor System.
22. KADS - Knowledge Acquisition and Development Method.
23. L3C - Laboratório de Conexionismo e Ciências Cognitivas.
24. LISP - List Processing.
25. MIT - Massachussets Institute of Tecnology.
26. NSPI - National Society for Programmed Instruction.
27. PGCC - Pós-graduação em Ciência da Computação.
28. PGEEL - Programa de Pós-graduação em Engenharia Elétrica.
29. PSSH - Physical Symbol System Hypothesis.
30. RBIE - Revista Brasileira de Informática na Educação.
31. RNA - Redes Neurais Artificiais.
32. RSL - Rigorous Approach to Industrial Software Engineering Specification Language.
33. RTM - Representational Theory of the Mind.
34. SE - Sistemas Especialistas.
35. SBC - Sociedade Brasileira de Computação.
36. STM - Short-Term Memory.
37. UFSC - Universidade Federal de Santa Catarina.

38. VDL - Viena Development Language.
39. VDM - Viena Development Method.
40. VDM-SL - Vienna Development Method Specification Language.
41. VVSL - VIP VDM Specification.
42. WWW - World Wide Web.

Lista de Símbolos

1. Sistemas formais: $\Gamma, \Pi, \Sigma, \Omega, \Xi, \Psi$
2. Alfabetos: Φ, Σ
3. Cadeias finitas: Φ^*
4. Cadeia vazia: ε
5. Comprimento de cadeia: μ
6. Teoria: Θ
7. Morfismo: \xrightarrow{f}
8. Função de entrada: Ψ
9. Conjunto de funções de saída: Λ
10. Função de transição de estados: Σ
11. Função de saída: δ
12. Função de transição de estados: λ
13. Função de próxima saída: η
14. Conjunto de saídas de um Autômato M: α, β, π
15. Função de próximo estado de M: ρ
16. Função de próxima saída de M: δ
17. Função de saída do próximo estado de H_p : δ

Lista de Publicações

1. ALMEIDA, M. A. F. ; BARRETO, J. M.; MACHADO, M. A . S. “Computação aplicada ao Ensino utilizando Inteligência Artificial”, *Pesquisa Naval - Suplemento Especial da Revista Marítima Brasileira*, Serviço de Documentação da Marinha, n.14, p. 81-96, Rio de Janeiro, outubro, 2001.
2. ALMEIDA, M. A. F.; BARRETO, J. M. “Modelagem Categórica de Sistemas Hipermídia como Autômato: Categoria dos Hipertômatos”, In: *Anais do XVI Simpósio Brasileiro de Sistemas Hipermídia e Multimídia - SBMIDIA'2001*, p. 227-228, Florianópolis, outubro, 2001.
3. ÁVILA, L. M., BRASIL, L. M., ALMEIDA, M. A. F. “Ensino de redes neurais e seu uso como ferramenta auxiliar ao profissional da área da saúde”, *SIARP'2001*, Florianópolis, outubro, 2001.
4. ÁVILA, L. M.; BRASIL, L. M.; ALMEIDA, M. A. F. “Proposal of Actions for Creation Of One Multimedia Enviroment to a Tutorial In Artificial Neural Networks”, In *VJMEDIMAG'2001*, Hanoi, Vietnam, 2001.
5. ALMEIDA, M. A. F.; BARRETO, J. M.; MACHADO, M. A. S. “Inteligência Artificial - uma aplicação ao Ensino”, In: *Simpósio Brasileiro de Pesquisa Operacional, SOBRAPO'2001*, Campos do Jordão, 2001.
6. ALMEIDA, Maria A F . ; BARRETO, Jorge M. “Ensino com Computador e Internet”, In: BARRETO, Jorge Muniz. (Org.), *Inteligência Artificial no Limiar do Século XXI*, 3ª. ed. Florianópolis, v. 1, p. 292-305, 2001.

7. ALMEIDA, M. A. F.; BARRETO, J. M. "Implementação de um Sistema Hipermídia como Autômato para Ensino de Inteligência Artificial. In: *Workshop de Teses e Dissertações em Informática em Educação do I Simpósio Catarinense de Computação*, Itajaí, agosto, 2000.
8. ALMEIDA, M. A. F.; BARRETO, J. M. "Modelagem de um Sistema Hipermídia como Autômato para Ensino de Inteligência Artificial" In: *XX Congresso Nacional da Sociedade Brasileira de Computação - SBC'2000*, v.1, p.1-8, Curitiba, 2000.
9. ALMEIDA, Maria A. F. ; BARRETO, Jorge M. "Inteligência Artificial e Ensino com computador". In: BARRETO, Jorge Muniz. (Org.), *Inteligência Artificial no Limiar do Século XXI*, 2ª. ed. Florianópolis, SC, v. 1, p. 232-246, 1999.
10. ALMEIDA, M. A. F.; BARRETO, J. M. "Formação continuada por Simulação em Ambiente Virtual usando Redes de Computadores" *Pesquisa Naval - Suplemento Especial da Revista Marítima Brasileira*, n.12, cap. XX, p. 271-287, Rio de Janeiro, 1999.
11. ALMEIDA, M. A. F. ; BARRETO, S. Q. ; CORRÊA, M. C. O. B & SOUZA, L. L. "Aprendizado Cooperativo utilizando Construtivismo e Redes de Computadores no Ensino a Distância" In: *Anais da 51ª Reunião Anual da SBPC*, Porto Alegre, julho, 1999.
12. ALMEIDA, M. A. F. ; BARRETO, J. M. & SANTOS, M. M. D. , "Um ambiente Computacional para Ensino de Redes Neurais Artificiais", In: *Anais da 51ª Reunião Anual da SBPC*, Porto Alegre, julho, 1999.
13. ALMEIDA, M. A. F. & BARRETO, J. M. "Apoio ao ensino de Redes Neurais Artificiais via computador", *XXII Congreso Argentino de Bioingeniería - SABI99*, Universidade de Favaloro, Argentina, junho de 1999, (CD-ROM(101/3)-(10998-ARGENDISK)).
14. ALMEIDA, M. A. F. & BARRETO, J. M. "Ambiente para ensino do projeto de Filtros Digitais no domínio do tempo: aplicação a sinais de eletrocardiografia",

In: *Anais do IV FNCTS - Fórum Nacional Ciência e Tecnologia em Saúde*, Curitiba, p. 407-408, outubro, 1998.

15. BARRETO, S. Q. ; SOUZA, L. L. FIALHO, F. P. & ALMEIDA, M. A. F. ,
“A vida pede licença para entrar na Escola”, In: *Anais da 49ª Reunião Anual da SBPC*, Belo Horizonte, p. 412, julho, 1997.

Resumo

Este trabalho visa aplicar métodos formais na Engenharia de Programas Educacionais Hipermedia, unificando as Teorias das Categorias, Autômatos. Para oferecer o suporte necessário à sua compreensão, o trabalho apresenta a evolução do uso das máquinas na Educação, os paradigmas da Ciência Cognitiva e os conceitos de sistemas formais que foram utilizados na modelagem dos sistemas aqui abordados. É apresentado, à luz dos princípios categóricos, um modelo teórico de um sistema Hipermedia como Autômato, denominado *Hipertômato*. Os modelos hipertômatos foram utilizados na construção de protótipos de ambientes de Ensino-aprendizagem Hipermedia, via redes de computadores, em diferentes domínios do conhecimento.

Palavras-chave: Engenharia de Programas Educacionais, Hipermedia, Categorias, Autômatos, Ciência Cognitiva, Hipertômatos.

HYPERTOMATA IN COMPUTERS ON EDUCATION

Maria Aparecida Fernandes Almeida

Electrical Engineer, Master in Electrical Engineer, Master in Computer Science

This work is presented as partial requirement for obtainance of a Doctoral degree in Computer Science Program of Federal University of Santa Catarina, Florianopolis, SC, Brazil, 2002.

Abstract

This work intends to apply formal methods in Hypermedia Educational Software Engineering unifying the Theory of the Categories and Automata. To offer the necessary support to its understanding, the work presents the evolution of the machines use on Education, the paradigms of the Cognitive Science and the concepts of formal systems that were used in the modeling of the systems approached here. It is presented, to the light of the categorical principles, a hypermedia theoretical model as Automata, denominated Hypertomata. The hypertomata models were used in the construction of prototypes of hypermedia web-based learning, in different domains of the knowledge.

Keywords: Educational Software Engineering, Hypermedia, Categories, Automata, Cognitive Science, Hypertomata.

Capítulo 1

Introdução

“Computer Science is an empirical discipline. We would have called it an experimental science, but like astronomy, economics, and geology, some of its unique forms of observation and experience do not fit a narrow stereotype of the experimental method. Nonetheless, they are experiments. Each new machine that is built is an experiment. Actually constructing the machine poses a question in nature; and we listen for the answer by observing the machine in operation and analyzing it by all analytical and measurement means available. Each new program that is built is an experiment. It poses a question to nature, and its behavior offers clues to an answer. Neither machines nor programs are black boxes; they are artifacts that have been designed, both hardware and software, and we can open them up and look inside. We can relate their structure to their behavior and draw many lessons from a single experiment.”

A. Newell & H. A. Simon, “ACM Turing Award Lecture”, 1976.

1.1 Motivação

O interesse da autora na concepção de sistemas de ‘Ensino com computador’ iniciou-se em 1995, durante seu curso de Mestrado em Engenharia Elétrica na UFSC [4]. Em 1996, a mesma desenvolveu no Grupo de Pesquisas em Engenharia Biomédica (GPEB) do Programa de Pós-graduação em Engenharia Elétrica (PGEEL) da Universidade Federal de Santa Catarina (UFSC), sob a orientação do Prof. Jorge Muniz Barreto, um ambiente computacional para Ensino de Projeto de Filtros Digitais no Domínio do Tempo [4], [7], [8], [9], [10], [11], [12], [18] como parte integrante de sua

dissertação de Mestrado em Engenharia Elétrica [4]. Neste mesmo ano, a autora desenvolveu, paralelamente, como trabalho de curso em uma das disciplinas deste Mestrado, um ambiente de Ensino de Redes Neurais inspirado nas Notas de Aulas de seus Professores Fernando Azevedo, Renato Garcia e Jorge Barreto.

Em 1997, motivada pelo Prof. Barreto e seus colaboradores¹, a mesma ampliou o sistema de Ensino de Redes Neurais. Em julho do mesmo ano, por ocasião do III Congresso Brasileiro de Redes Neurais (III CBRN), o sistema foi disponibilizado em uma máquina local para uso dos participantes do III CBRN. Incentivada pelo interesse das pessoas devido a originalidade do tema (na época não existia programas hipertextos similares em língua portuguesa), ela resolveu melhorar seu conteúdo e interface.

Várias versões foram desenvolvidas (entre 1995 e 1999) com auxílio das críticas e sugestões de professores e estudantes de graduação e pós-graduação em Engenharia Elétrica e Ciência da Computação da UFSC. O Ciclo de Vida deste sistema terminou quando a autora percebeu que, para ser efetivamente usado como recurso pedagógico no Ensino de Redes Neurais, tornava-se necessário uma adaptação tão complexa que era melhor finalizar seu Ciclo de Vida. Por mais recursos atrativos que fossem colocados, não valeria a pena todo o esforço dispensado com a ferramenta de autoria pois ele tinha sua própria limitação: era mal concebido...

Em 1998, a autora iniciou seu Mestrado em Ciência da Computação na UFSC desenvolvendo a modelagem de um ambiente de Ensino Hiperídia como Autômato [5]. Em 1999, ela observou que esta modelagem poderia ser aplicada ao seu sistema morto, ou seja, nasceria do conteúdo daquele, um novo sistema que já na sua concepção poderia ter grandes chances de sobrevivência e de motivar a concepção de novos sistemas. O problema era como poderia ser feita a atualização do conteúdo já que o programa original festejava seu quinto ano?

Dessa feita, a autora contou com o auxílio do Prof. Mauro Roisemberg do INE que, ministrando aulas de Inteligência Artificial no curso de Pós-graduação em Ciência da Computação (PGCC), solicitou uma avaliação do programa entre seus alunos. Esta valiosa colaboração permitiu a modelagem de uma versão independente da plataforma para uso em rede de computadores. Assim, aliando as solicitações dos estudantes ao objeto de estudo de sua pesquisa, a autora criou o grafo de implementação de um ambiente de Ensino de Redes Neurais [13], [14], [21], [22], seguindo a modelagem de um sistema Hiperídia como Autômato, conforme proposto na sua dissertação de Mestrado em Ciência da Computação [5].

¹Dentre os colaboradores estavam os alunos dos curso de Bacharelado em Ciência da Computação do Departamento de Informática e de Estatística (INE) da UFSC que, na época, utilizaram o ambiente como apoio na disciplina de Inteligência Artificial.

Considerando os bons resultados obtidos nas suas pesquisas anteriores [4], [5], Almeida iniciou, em 1999, o curso de Doutorado em Ciência da Computação no PGCC-UFSC, procurando alternativas para melhoria da qualidade dos programas educacionais, principalmente no que se refere a concepção do sistema [15], [16].

No primeiro semestre de 1998, a autora desenvolveu um estágio de docência² por reconhecer a importância desta atividade na sua formação acadêmica e profissional. Além disso, para atingir seus objetivos, aplicando sua pesquisa em turmas de alunos reais, a mesma efetuou, durante o ano de 2000, dois estágios de docência consecutivos³ em disciplinas oferecidas pelo INE no Laboratório de Informática do Centro de Ciências Agrárias (CCA) da UFSC. Estas disciplinas destinam-se ao Ensino de conceitos introdutórios de Computação para Ciências Agrárias e Engenharia de Aquicultura. Como recurso didático, afim de avaliar os conceitos desenvolvidos nesta tese, a autora concebeu um sistema de apoio ao Ensino via rede para que fosse utilizado e avaliado pelos alunos do CCA. Os últimos estágios, embora ultrapassassem o número de créditos recomendados nos curso de mestrado e doutorado, efetuaram-se visando a grande utilidade de poder “testar em campo” suas pesquisas sobre “Ensino com computador”.

A autora considera importante “Ensinar com o computador” e não somente desenvolver ferramentas puramente teóricas. Assim, foi para a sala de aula usando como recurso pedagógico as implementações de ambientes computacionais de Ensino, desenvolvidos seguindo a modelagem proposta nesta tese, verificando erros e acertos, observando as reações dos alunos, suas preferências, ouvindo opiniões, críticas e sugestões no dia a dia; ou seja, vivenciando, na prática, o que denominam de “mitos e verdades da Informática na Educação”.

Em 2001, a autora ampliou suas pesquisas analisando o perfil de turmas reais de alunos em diversos cursos de graduação da UFSC. Contando com a colaboração de professores e alunos de várias disciplinas de “Introdução à Computação” (ICC) foi possível delinear o perfil dos ‘clientes’ através da ‘análise de oportunidades’⁴.

Um dos motivos da escolha de alunos de diversas áreas dos cursos da UFSC foi observar o problema da aceitabilidade do computador em turmas que não estão familiarizadas com o mesmo, esta escolha reflete a realidade brasileira na qual a maioria das pessoas não sabe manipular o computador. Os resultados obtidos confirmaram a necessidade de formalização no desenvolvimento dos programas educacionais pois

²Disciplina INE-5371 - Inteligência Artificial do Curso de Bacharelado em Ciência da Computação do INE-CTC-UFSC.

³Disciplinas INE-5214 - Introdução à Computação e Informática Agrícola e INE-5220 - Introdução à Computação.

⁴É utilizada aqui a terminologia da primeira fase da Engenharia de Programas.

os aspectos psico-pedagógicos envolvidos na concepção e manipulação de sistemas de Ensino-aprendizagem têm sido muito bem explorados por valiosos trabalhos na área ao longo dos anos. Assim, vale a pena ressaltar que esta tese evidencia o valor de toda a pesquisa desenvolvida seguindo os paradigmas educacionais. Todavia, este assunto não é abordado aqui devido a sua extensão e por fugir dos objetivos aqui propostos.

A autora acredita que deve existir um contato direto entre alunos e professores. O computador é uma ferramenta poderosa para ampliar as capacidades intelectuais mas não pode substituir o professor, como proposto em vários sistemas de Ensino virtual. O computador deve servir para dinamizar o Ensino, usando a metáfora da Apple [25] “Rodas para a mente”; ampliar a capacidade de raciocínio e permitir que se percorra distâncias.

A ferramenta formal, aqui proposta para construção de sistemas hipermídia, pode ser estendida para o desenvolvimento de outras aplicações (comércio eletrônico, Ensino à distância, etc.). A aplicação de seus conhecimentos na concepção de ambientes de Ensino-aprendizagem deve-se, em grande parte, ao contato da autora com pessoas da área de Informática na Educação, no local onde é publicada a RBIE⁵, suas experiências anteriores (com trabalhos já apresentados na área [4], [5], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [28], [29], [54], [97]), como Pesquisadora⁶ e Professora Universitária, interessada nas novas tecnologias de Ensino e na qualidade do mesmo, serviram de forte motivação para o desenvolvimento deste trabalho.

⁵Revista Brasileira de Informática na Educação publicada pela Sociedade Brasileira de Computação (SBC).

⁶A autora é pesquisadora participante do Laboratório de Conexionismo e Ciências Cognitivas do CPGC-INE-UFSC, onde desenvolveu o Projeto ICEM - Inteligência Computacional no Ensino com Multimídia, do qual pertence o estudo desta tese, sob a orientação do Prof. Jorge Muniz Barreto.

1.2 Objetivos

1.2.1 Objetivo Geral

Ensinar com o computador, além de outros aspectos, envolve a concepção de um programa. O desenvolvimento de programas é base da Engenharia de Programas (EP)⁷.

A EP [216] passou por várias fases. Inicialmente, nos anos 50, 60 e 70, a preocupação era com a correção lógica de programas. Já nos anos 60, chegou-se a conclusão que fazer um programa realístico correto era um problema tão complexo que se tornava impossível. A EP voltou-se, então, para a abordagem construtiva de programas, ou seja, o delineamento de passos sucessivos para se evitar erros de semântica.

Os primeiros métodos de especificação formal foram introduzidos por Jackson [117], desenvolvidos para construção correta de programas em COBOL. Ao método de Jackson seguiram-se uma série de métodos formais de linguagens de especificação que, por serem sintáticas, ofereciam uma maneira precisa e sistemática de construção de programas, evitando-se erros [50], [66], [82], [139], [218]. Ainda nos anos 60, a IBM [138] de Viena criou o projeto VDM ("Viena Development Method"), que foi o método de desenvolvimento do compilador da linguagem PL-1 ("Programming Language 1") [139]. Neste projeto, foi desenvolvida uma série de ferramentas formais, que posteriormente foram generalizadas com a utilização de estruturas algébricas (grupos, anéis, corpos, etc.), surgindo então a VDL ("Viena Development Language").

Diversas linguagens de especificação formal foram inspiradas no VDM. Dentre elas destacam-se: a VDM-SL ("Vienna Development Method Specification Language") [151], a RSL ("Rigorous Approach to Industrial Software Engineering Specification Language") [103], a VVSL (VIP VDM Specification) [150] e a VDM++ (VDM orientada a objetos) [73].

A Teoria das Categorias [26], [211], foi introduzida como linguagem de especificação de programas no final dos anos 60. Atualmente, existe uma corrente da EP, muito rica, que são os métodos formais [130] e neles se inclui a Teoria das Categorias.

Existe muita preocupação com a melhoria da qualidade de sistemas de Ensino computadorizados através da utilização de diversos paradigmas Educacionais oriundos da Psicologia. Entretanto, existe pouco ou nenhum enfoque na concepção desses programas em termos computacionais.

⁷A "IEEE Computer Society" publica um periódico totalmente dedicado a esta área - a "IEEE Transactions on Software Engineering".

Considerando que um programa de Ensino com computador é um programa que possui um Ciclo de Vida [216], este trabalho pretende como objetivo geral:

- Utilizar Métodos Formais para aplicação na Engenharia de Programas Educacionais Hipermedia.

1.2.2 Objetivos Específicos

Para atender o objetivo geral, mencionado anteriormente, são definidos os seguintes objetivos específicos:

- Descrever a evolução das máquinas de Ensino que possuem histórico pouco documentado;
- Efetuar um estudo comparativo entre as teorias de Aprendizado Animal e o Aprendizado de Máquina, introduzindo o Aprendizado à luz da Teoria das Categorias;
- Apresentar conceitos rudimentares de Sistemas formais e Teorias das Categorias para aplicá-los na concepção de Sistemas de Ensino com computador;
- Apresentar o Modelo Teórico de um Sistema Hipermedia como Autômato (denominado Hipertômato) e sua unificação com a Teoria das Categorias;
- Mostrar as implementações desenvolvidas com Hipertômatos, utilizadas na prática em Sistemas de Ensino, em diferentes domínios;
- Observar o problema da aceitabilidade do computador de pessoas que não estão familiarizadas com o mesmo utilizando turmas de alunos reais;
- Apontar alternativas da utilização das técnicas formais da Computação para melhoria da qualidade de Sistemas Ensino com computador.

1.3 Estrutura do trabalho

Para atender os objetivos propostos, este trabalho está estruturado conforme descrito a seguir:

O Capítulo 1 apresenta as motivações e a justificativa para o desenvolvimento, os objetivos e a estrutura do trabalho.

O Capítulo 2 apresenta a evolução do uso das máquinas no Ensino, enfatizando o Computador como uma poderosa ferramenta para ampliar as capacidades intelectuais humanas.

No Capítulo 3 é apresentado um estudo comparativo sobre o Aprendizado Animal e o Aprendizado de Máquina. São abordados a Taxinomia do Aprendizado, os Paradigmas da Ciência Cognitiva e o Aprendizado sob Linguagem Categórica.

O Capítulo 4 mostra os conceitos elementares dos Sistemas Formais focalizando os conceitos rudimentares da Teoria da Categorias necessários à compreensão do presente trabalho.

No Capítulo 5 são descritos os conceitos básicos da Teoria dos Autômatos e de Sistemas Hipertexto (ou hipermídia) como Autômato. A unificação destas Teorias gerou o modelo, aqui denominado Hipertômato, tratado em linguagem categórica.

O Capítulo 6 mostra as implementações de sistemas de Ensino, utilizando o conceito de Ciclo de Vida, desenvolvidos em diferentes domínios. São apresentados os resultados obtidos na avaliação de um sistema modelado como Hipertômato, através de uma pesquisa em campo feita com alunos de turmas reais na UFSC.

No Capítulo 7 são apresentadas as considerações finais sobre o trabalho desenvolvido.

Capítulo 2

Máquinas para Ensinar...

*“Time present and time past
are both perhaps present in time future
and time future is contained in time past ...”*

T. S. Eliot, “Burnt Norton”.

Quando as pessoas começaram a utilizar máquinas para Ensinar? Quais foram as precursoras do computador no Ensino? Quais as relações entre as estratégias utilizadas no passado e as atuais? Este capítulo visa mostrar um breve histórico das máquinas utilizadas para Ensinar e discutir as possibilidades do computador no processo de Ensino-aprendizagem.

O computador talvez seja a ferramenta mais interessante que o ser humano já construiu para contribuir na ampliação de suas capacidades intelectuais e cuja vocação é o armazenamento e manipulação de informação. Não é de se estranhar portanto, que desde os primeiros dias dos computadores, tenha aparecido um grande interesse no uso dessas máquinas no Ensino.

Inicialmente, muitos acreditavam que a utilização do computador faria uma revolução total do processo de Ensino. Isto leva à seguinte reflexão:

- Houve, estará havendo, haverá tal revolução ?
- Haverá assuntos mais adaptados ao Ensino usando computadores?
- Qual o melhor modo de usar o computador?

Grande parte das observações contrárias à utilização de modernas tecnologias na Educação não são devidas a tecnologia em si, mas pelo uso que se faz dela. Num contexto geral, é preciso desmitificar o computador nas Escolas, é preciso mostrar aos professores como utilizá-lo e saber como ampliar as suas capacidades. É preciso mais: valorizar o homem à máquina. Torna-se necessário aplicar métodos de Ensino que valorizem a construção de conhecimentos do educador e do educando e através destes preceitos pode-se distribuir, num sentido amplo e social, os conhecimentos.

O uso das máquinas para Ensinar não é algo recente. Diversos modelos e estratégias foram utilizadas, até o uso do computador com implementações através de técnicas de Inteligência Artificial (IA), Realidade Virtual (RV) e Redes de Computadores. A IA aliada às redes de computadores, são poderosos recursos da Computação para melhoria da qualidade dos programas de Ensino. As aplicações em IA têm como objetivo modelar o conhecimento humano através de esquemas computacionais, na pretensão de que “imitando-se algo sabidamente inteligente, obteremos algo também inteligente”. A nova geração de sistemas inteligentes está sendo construída com o desenvolvimento de sistemas híbridos que constituem um novo campo da Inteligência Aplicada.

A web pode ser utilizada como uma ferramenta pedagógica em si ou como suplemento de outras formas de instrução. Entretanto, a complexidade deste ambiente aberto deve ser considerada cuidadosamente.

“... Como nos mantermos os humanos atrás do computador visível? Existe o perigo de que professores e instrutores de cursos comecem a enfraquecer? Como podemos impedir isto? É desejável que impeçamos tal coisa? ...” [81]

O interesse nas Máquinas para Ensinar remonta da Revolução Industrial estendendo-se ao princípio do século XX, com um volumoso êxodo de camponeses para as cidades em busca de novas perspectivas de vida. Estes trabalhadores teriam que receber então, uma formação rápida. Entretanto, haveria uma escassez de professores capazes de os formar, provocando, assim, vários empreendimentos para automatização do Ensino. Precisava-se de teorias de Ensino para sustentar tais idéias, que seriam providas pelas teorias da Psicologia.

Segundo Bordeleau [45], a primeira patente conhecida de uma máquina para ensinar foi feita pelo americano H. Chard em 1809, denominada “Mode of Teaching Reading” para Ensino de leitura.

2.1 Máquina de Pressey

Em 1913 nascia o aprendizado individualizado seqüencial através do trabalho de Edward L. Thorndike (1874-1949)[107], do Columbia University Teachers College. Thorndike [107] propôs a individualização da instrução e inventou a noção do prévio: um estudante não deveria passar para uma próxima lição sem que tenha feito os exercícios e aprendido a lição precedente. É a abordagem seqüencial: o estudante não deve ir para a página seguinte de um livro se ele não obtiver sucesso nos exercícios de uma página prévia.

Assim, em 1924, surge a primeira máquina para Ensino seqüencial, o Tutor Drum. Esta máquina idealizada por Sidney Pressey, professor de Psicologia da Educação na Universidade do Estado de Ohio, era dirigida a um grande grupo de estudantes como sendo um sistema automatizado de questões. Pressey apresentou a máquina na Conferência Anual da “American Psychological Association”, apoiando-se nos princípios educacionais de Thorndike: o aluno não poderia ir para a instrução seguinte sem que tivesse sucesso prévio. A máquina continha um teclado chaveado para busca de respostas de múltiplas escolhas e uma janela, na qual eram colocadas as questões. Isto prevenia que o aluno seguisse e o mesmo recebia uma realimentação imediata no caso de insucesso. Em 1932, Pressey, publica um artigo na “School Magazine Society”, preconizando que aconteceria uma revolução real na educação com uso destas máquinas...

2.2 Máquinas para Ensino: fonógrafo e televisão

Segundo Lachance apud Bordeleau [44], em 1909 foi utilizado numa escola pública americana (Milwaukee, EUA), o primeiro fonógrafo como recurso pedagógico. Dois anos depois, a companhia de discos RCA Victor lançou o primeiro disco gravado para o Ensino. Em seguida, 1912, vai ao ar a primeira estação de rádio educativa dos Estados Unidos, como uma iniciativa da Universidade do Estado de Ohio, que começou a distribuir cursos para seus estudantes. A Universidade de Wisconsin e a Universidade do Estado de Iowa repetiram o feito em 1916 e 1919, respectivamente. Mas a concessão de licença para funcionamento de uma estação de rádio educativa, foi para a Universidade da Cidade de Salt Lake, em Utah, no ano de 1921.

Em janeiro de 1933 tem-se o início da primeira televisão educativa americana [62]. A Universidade de Iowa difunde através da estação W9XK o primeiro programa de televisão que apresenta uma breve conferência da universidade sobre ela mesma, com um solo de violino, uma lição de desenho à mão livre e um excerto de uma peça teatral. Desse feito foram observadas as múltiplas possibilidades desta

técnica audiovisual no Ensino. Assim, em 1952, a Comissão Federal Americana de Comunicação alocou à televisão uma cadeia de 242 canais educacionais [44].

2.3 Máquinas de Skinner

Em 1954, Skinner [202], [203] introduz o *Behaviorismo* influenciado pelos trabalhos de Pavlov no condicionamento de animais e nas idéias de Thorndike sobre treinamento na educação. Mais detalhes sobre os trabalhos de Pavlov e Thorndike podem ser obtidos em Hergenhahn [107].

A proposição educativa de Skinner é a colocação de reforços para promoção da aprendizagem. Sua pretensão era a de que qualquer aluno pudesse aprender “elementos do conhecimento” com a ajuda de uma técnica que denominou *aprendizado programado*. Os elementos mais simples eram colocados acessíveis aos alunos através de unidades (ou “frames”).

O aluno então, assimilaria tais elementos em ritmo próprio, num processo de perguntas e respostas seguidas de reforços.

Em 1958, Skinner, cuja foto pode ser vista na Figura 2.1, imaginou uma máquina que fosse capaz de conter os ensinamentos programados e entregar tais elementos em pequenas porções de conhecimento, conforme a progressão das respostas dos estudantes. Ele inicia o aperfeiçoamento da máquina de Pressey introduzindo o conceito de *programação linear*: os conhecimentos são apresentados sucessivamente e os alunos não passam à unidade (“frame”) seguinte sem que respondam corretamente a questão (estímulo) que lhes é apresentada. Suas pesquisas foram financiadas por militares e indústrias que necessitavam de formação de mão de obra rápida.



Figura 2.1: B. F. Skinner . Fonte: UPI- The Bettmann Archive

As Figuras 2.2, 2.3 e 2.4 mostram algumas dessas máquinas que foram produzidas na época.

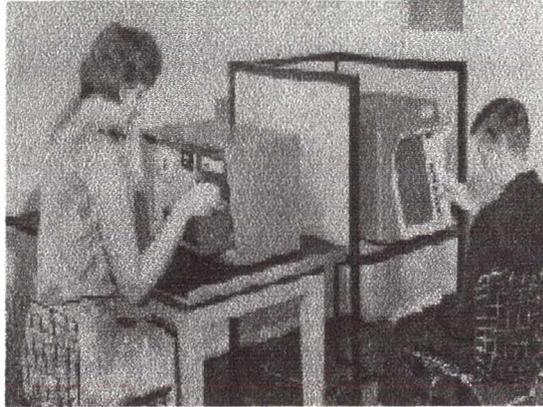


Figura 2.2: Discoverer: programas lineares. Fonte: [44]



Figura 2.3: Máquina Devereux na França. Fonte: [45]

Skinner [203], descreveu a operação de uma máquina para Ensino como:

“... A frame of textual material appearing in the square opening is incomplete: in place of certain letters or figures there are holes. Letters or figures can be made to appear in these holes by moving slides (a keyboard would be an obvious improvement). When the material has been completed, the student checks his response by turning a crank. The machine senses the settings of the sliders and, if they are correct, moves a new frame of material into place, the sliders returning to their home position...”

“The student sees printed material in the large window at the left. This may be a sentence to be completed, a question to be answered, or a problem to be solved. He



Figura 2.4: Auto-Tutor (EUA)

writes his response in an uncovered portion of a paper strip at the right. He then moves a slider which covers the response he has written with a transparent mask and uncovers additional material in the larger opening...".

Skinner acentua mais o programa instrucional do que a máquina para ensinar mas logo percebe que sua abordagem educacional não poderia ser aplicada eficientemente numa sala de aula, porque havia uma falha particular que ainda hoje é um grande obstáculo no uso dos computadores nas escolas: a formação adequada para se lidar com a tecnologia. Assim, ele sentiu que seu trabalho poderia ser adequado a graus elementares mas em sistemas mais complexos (Ensino médio e superior), seria necessário uma máquina mais poderosa, e o principal: professores para manipulá-la.

A abordagem "Skinneriana" para desenvolvimento de programas de computador sofreu (e sofre) muitas críticas pela metodologia aplicada, que não proporciona a aprendizagem por exploração e descoberta, mas historicamente suas pesquisas (em sua maioria, desenvolvidas nos anos 50) têm grande valor científico.

Skinner descreve como a máquina poderia explicar erros comuns quando o estudante comete um engano. A cada erro o estudante perfuraria um buraco no papel ao lado do erro para que o professor pudesse revisar o progresso. Ele concluiu que:

"Exploratory research in schools and colleges indicates that what is now taught by teacher, textbook, lecture, or firm can be taught in half the time with half the effort by the machine of this general type...".

2.4 Máquina de Crowder

Em 1959 Crowder apud Bordeleau [44], um instrutor da Força Aérea Americana, critica a programação linear de Skinner: seus erros - segundo Crowder, seriam inevitáveis no decorrer da instrução mesmo que se tivesse mecanismos para corrigi-los. Assim, ele propôs o conceito de *programação ramificada*, denominada de programação intrínseca [44].

Na programação ramificada, desde o início do programa, devem ser consideradas as diferenças individuais para se adaptar a instrução ao aluno. Crowder propôs, então, uma máquina que se ramificava afim de que o aluno atingisse conhecimentos mais complexos com a progressão de suas respostas. A máquina, muito sofisticada para a época, conteria rolos de filmes nos quais seriam fixadas seqüências de múltiplas instruções. Conteria, ainda, consoles e botões de relés que, juntos, permitiriam aos estudantes responderem às questões propostas. Mas a cada novo curso, tornava-se necessário recarregar a máquina, uma operação complexa, o que limitou sua utilização.

Pode-se dizer que o conceito de Crowder sobre programação ramificada é um ancestral dos sistemas Hipermídia no Ensino. A implementação visual da Máquina de Crowder são os denominados “grafos crowderianos”. Estes grafos nada mais são do que os grafos de hipertexto. A matéria era dividida em nós de conhecimentos e estes nós de conhecimentos eram ligados através de “links” semânticos. Uma lição era percorrer estes “links” semânticos e receber o conhecimento através desses nós.

O hipertexto nasceu nos anos 40, mas não se pensava na sua utilização no Ensino enquanto os grafos crowderianos destinavam-se a este fim.

2.5 Máquina de Pask

Gordon Pask [102], um estudioso inglês da Cibernética [189], [219], acreditava que a abordagem de Skinner impunha ao aprendiz um padrão muito rígido de comportamento. Assim como Crowder, ele acreditava que poderiam ser estabelecidas seqüências de instruções em outro modelo que não fosse a linearidade. Seu trabalho, após 1954, foi dirigido ao conceito de máquinas para ensinar que poderiam se ajustar ao estilo de treinamento de cada aluno.

No entender de Pask, o modo de intercâmbio com a máquina deve ser de natureza mais cooperativa do que diretiva. Deveria assim, ser estabelecida uma série de diálogos (produzidos entre os alunos e a máquina) afim de que a mesma adaptasse suas respostas aos alunos e não ao contrário. Pask orientou sua atenção mais nas respostas positivas dos alunos do que nas negativas; nos tipos de erros cometidos; nos

tempos das respostas; com intuito de variar os graus de dificuldade das perguntas em função destes dados.

Pask estava mais interessado em modelar o Ensino sobre as respostas do aluno do que no conhecimento do professor, o que constitui uma ruptura em relação ao modelo Skinneriano clássico e o modelo de Crowderiano que se limitava a ramificações que continham algumas sucessões de instruções lineares e fechadas. Em 1959, Pask apresenta sua denominada *Teoria da Conversação*, para a qual seria necessária uma máquina eletrônica muito rápida (computador?) para estabelecer diálogos.

A Figura 2.5 mostra uma máquina, do tamanho de uma máquina de escrever, que poderia ser abastecida com material didático, denominada DIDAK . Foi produzida pela Rhemm-Califone Corporation no final dos anos 50 [102].



Figura 2.5: Máquina Didak Fonte: [221]

A euforia ultrapassara os limites do “bom senso”... pois certa vez um educador americano, segundo Wilson [220], descreveu-a como:

“... A professora ideal que sempre diz a coisa certa no momento exato, ajusta... a matéria ao... aluno, tem uma paciência... sem fim, e nunca se enfada.”.

A Figura 2.6 mostra uma tira de papel na janela da DIDAK intercalando fatos com sentenças, que o aluno completa no espaço adequado. Os defensores desta máquina afirmavam que o aprendiz participava ativamente do processo de aprendizagem, trabalhava em seu próprio ritmo e saberia durante todo o tempo, seu progresso, ou seja, todas as práticas educacionais que seriam ideais. Acreditava-se também que essa máquina teria um modo de Ensino individualizado e a vantagem

de ser produzida em massa. Entretanto, ela não fazia mais do que apresentar o material previamente colocado. Como poderia alguém dizer que substituiria um ser humano? Ainda hoje, muitos acreditam que “cérebros eletrônicos”, (como eram denominados os computadores antes do anos 80), dotados de “inteligência artificial” irão substituir seres humanos no Ensino. O que dizer a eles? A resposta pode ser simples - o computador é apenas uma ferramenta para ampliar as capacidades intelectuais, assim como um martelo (para fixar um prego na parede) ou um avião (para chegar mais rápido a um lugar distante) ampliam as capacidades físicas.

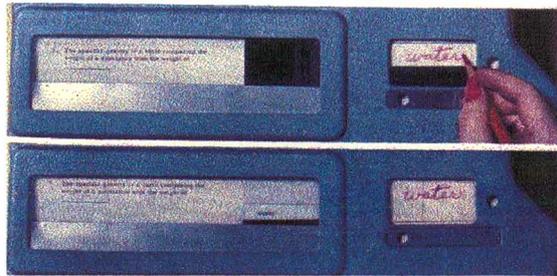


Figura 2.6: Janela da Máquina Didak Fonte: [221]

Em 1962, acontece uma conferência na Randolph Air Forces Basis [193] do Texas. Este encontro reuniu uma série de especialistas do Ensino programado na formação militar. Desta reunião nasce então a “National Society for Programmed Instruction (NSPI)”. Ao mesmo tempo surge o “Journal for Programmed Instruction” com diversas publicações na área, dentre as quais destaca-se a de Susan Markle em 1964. Markle publica um manual de Ensino programado intitulado “Good Frames and Bad” [44]. O sucesso desta publicação foi tanto que tornou-se o manual para autores do Ensino programado porque expressava os princípios de desenvolvimento de tal Ensino.

2.6 Máquina para Ensinar: O computador

Os primeiros programas educacionais foram influenciados pela teoria “Behaviorista” [202]. Neste paradigma educacional o Ensino ou instrução é um processo de estímulo-resposta e existe uma grande preocupação na medição da aprendizagem [182]. As idéias eram influenciadas pelas teorias mecanicistas de Pavlov, Thorndike, Watson [107] e Skinner [202]. Passo a passo, são desenvolvidas alternativas ao Ensino programado fundamentado na Teoria Construtivista ¹ de Piaget [175] [177]

¹Mais detalhes são mostrados na seção 3.1.

que tem grande efeito na Educação. Para Piaget, a educação ótima envolve suas mudanças nas trocas de experiências para o estudante a fim de que os processos duais de assimilação e acomodação promovam o crescimento intelectual. Para criar tal experiência, o professor deve conhecer o nível de funcionamento da estrutura cognitiva de cada estudante. Dentre as alternativas da utilização do computador no Ensino destacam-se os trabalhos pioneiros de Seymour Papert [173] do MIT, através da introdução de uma linguagem de programação de computadores especialmente desenvolvida para a Educação, a linguagem LOGO [173].

Na aplicação da Informática na Educação são encontrados inúmeros trabalhos Construtivistas [68], [69], [77], [78], [173], dentre os quais muitos utilizam a IA [46], [178], [179]. A IA Construtivista pode ser capaz de auxiliar na adaptação de modelos tradicionais e estratégias instrucionais para projetar sistemas de ensino nos quais as experiências educacionais devam ser construídas em torno da estrutura cognitiva do estudante. Os construtivistas evitam a formulação de modelos para o processo de ensino-aprendizado. Os modelos construtivistas não são sistemas construídos. Assim, suas teorias são mais parciais e por tentativas, procurando conectar o conhecimento com a prática, dependendo do contexto. Embora a tecnologia seja uma parte integrante da Informática Educativa, qualquer programa que pretenda êxito tem que focar mais as necessidades instrutivas dos estudantes do que a própria tecnologia. É essencial considerar os fatores culturais e socio-econômicos, os interesses e as experiências, os níveis educacionais e a familiaridade com métodos utilizados.

No Brasil, o uso do computador na Educação é algo recente se comparado a países como EUA e França. Todavia, desde 1973, o Brasil realiza experimentos com computadores na Educação cujo Estado pioneiro foi o Rio Grande do Sul. O avanço tecnológico nas áreas de comunicação e Informática para ser utilizada em Educação, data oficialmente de 3 de outubro de 1979, quando foi criada a SEI (Secretaria Especial de Informática) como órgão complementar do CSN (Conselho de Segurança Nacional), assumindo as funções acumuladas pela CAPRE (Comissão de Coordenação das Atividades de Processamento Eletrônico), como a formação de recursos humanos para o setor de Informática [67].

Em março de 1980 a SEI cria a Comissão Especial de Educação, um segmento de apoio ao MEC, para assumir o papel de gerador de normas e diretrizes do campo que se abria para a Educação. Acontecem, a partir daí, os Seminários Nacionais de Informática na Educação (I - agosto de 1981, em Brasília, II - agosto de 1982, na Bahia e III e I - Internacional, novembro de 1986, no Rio Grande do Sul).

No início de 1983, os membros da Secretaria reuniram-se para a elaboração do projeto EDUCOM : Informática na Educação, que foi aprovado no mesmo ano. Em fins de 1984, por delegação atribuída pelos membros do EDUCOM, o Centro

de Informática Educativa (CENIFOR² que foi proposto em 1983 pela Fundação Centro Brasileiro de TV-Educativa), assume a responsabilidade pela coordenação e supervisão técnica da execução do projeto EDUCOM. A partir de 1984, o projeto EDUCOM procurou a implantação de centros-piloto, Minas Gerais (UFMG), Pernambuco (UFPE), Rio Grande do Sul (UFRGS) e São Paulo (UNICAMP). Uma ótima leitura sobre este assunto pode ser encontrada nos trabalhos de Fernandes [77], [78].

Atualmente, a ligação entre o Ensino e o computador pode ser vista em um contínuo de facetas, indo desde o uso do computador em ambientes de Ensino-aprendizagem para facilitar diversas tarefas usuais, até o fato do computador tomar parte ativa e total no processo. No primeiro caso o computador age como elemento de modificação de uma cultura. O Ensino proporcionado é dependente da modificação desta cultura e é já fato corrente em muitas instituições de Ensino. Um fator importante nesta modificação de cultura é o fato de existir microcomputadores com capacidades suficientes para uso em tratamento de textos, preparação de gráficos, criação de pequenas base de dados, planilhas, etc.

No outro extremo tem-se o computador tomando todas as iniciativas do processo de transmissão de conhecimentos, desde a apresentação do assunto até a verificação de conhecimentos. Em suma, o espectro de utilizações de computador no Ensino pode ser classificado, entre outros, como:

- agente modificador do ambiente de vida do aluno;
- jogo educativo;
- aparelho de laboratório;
- enciclopédia;
- interlocutor pedagógico;
- outras...

Os jogos, popularizados com a disseminação dos microcomputadores, produzem freqüentemente uma certa aprendizagem como efeito colateral. Por exemplo, um simulador de vôo permite introduzir noções básicas de pilotagem, outros apuram reflexos, outros desenvolvem memória. Um ponto alto nos jogos é a motivação, dependendo do filtro do aprendiz. Um ponto negativo é que o rendimento da aprendizagem pode diminuir.

²O CENIFOR foi extinto em 1987.

Como aparelho de laboratório o computador permite experiências usando o conceito de simulação (no sentido de experiências com modelos em geral). Há uma grande liberdade por parte do aluno que é livre, de uma certa forma na escolha de suas experiências.

A linguagem LOGO é um exemplo simples, na qual o computador simula o movimento de uma tartaruga que deixa atrás de si um rastro e que se move obedecendo a comandos.

O uso de enciclopédias é feito quando o computador atua como armazenador de conhecimentos que podem ser consultados segundo as necessidades dos alunos. Tal uso pode ir desde uma implementação de banco de dados para consultas até um esquema capaz de, usando conhecimentos enciclopédicos, efetuar deduções.

O uso como interlocutor pedagógico é a forma geralmente pensada quando se fala em Ensino com computador e é aquele em que o computador substitui, ao menos durante intervalos de tempo, um ser humano. Este modo é especialmente útil na formação contínua após os estudos em estabelecimentos de Ensino, embora seu uso durante o Ensino formal possa ampliar suas possibilidades.

Alguns usos do computador no Ensino primário e secundário, podem ser comprovados dos seguintes modos:

- práticas e exercícios: é bastante usado pois os programas são geralmente simples. O programa apresenta exercícios e verifica se as respostas são boas;
- tutoriais;
- ferramentas de auxílio ao Ensino;
- simulações.

Segundo Bordeleau [45], a primeira aplicação do computador na instrução data de 1950. O programa denominado *The Whirlwind* era um simulador de vôo para formação de pilotos em combate desenvolvido no M.I.T., Boston, por Ken Olsen e Robert Everett. A partir disso, Olsen fundou a Digital Equipment Corporation (DEC) que tornou-se famosa pela produção dos minicomputadores da série PDP [55], [119]. Everett vai, então, para a Mitre Corporation e finaliza o primeiro sistema de instruções por televisão assistida por computador (TICCIT).

Em 1957, Simon Ramo, engenheiro e industrial americano, apresenta uma visão futurística dos computadores na educação. Ramo acreditava que as classes iriam ser automatizadas por completo. O computador seguiria os rastros da aprendizagem do aluno automaticamente através da interação deste com botões elétricos. Nesta visão, não haveria mais lugar para professores. Esses deveriam ser reciclados

para programar estas máquinas. Ramo propôs o conceito de “Computer Managed Instruction (C.M.I.)”, isto é, Instrução Administrada por Computador. Em outras palavras, Ensino gerado por computador, que necessitava de bancos de programas e banco de dados de alunos que só o computador poderia gerar. Sob esta visão o computador poderia:

- testar e medir a aprendizagem dos alunos e preservar os resultados;
- diagnosticar as dificuldades deles e prescrever um Ensino corretivo;
- produzir um relatório sobre o progresso do aluno.

Os “idealizadores” deste sistema preconizavam que os professores se livrariam da administração de treinamentos e poderiam dedicar-se à preparação das lições. Esta visão, tinha uma parte extremamente delicada pois exigia a reciclagem dos professores para programar tais máquinas. Tal exigência não é pois semelhante à atual, na qual se exige que um professor entenda a tecnologia para usá-la? Nota-se que as coisas não mudaram muito nestes últimos cinqüenta anos...

Em 1959, a IBM desenvolveu, para seus próprios propósitos de formação de pessoal, um sistema de Ensino assistido por computador. Esse sistema foi experimentado em alunos das escolas públicas do Estado de Nova York, EUA, no aprendizado de Matemática Binária e proposto por Rath, Anderson e Brianerd, pesquisadores do “Teaching Machine Project” no “Watson Research Center” da IBM. Os alunos utilizavam terminais (teletipos) ligados, através de linha telefônica, aos computadores da IBM. Após este projeto os pesquisadores finalizaram o primeiro sistema de autoria computacional, o “CourseWriter”, que funcionou alguns anos depois, no primeiro minicomputador dedicado ao Ensino, o IBM 1500.

Na Universidade de Illinois, no início dos anos 60, Stolhuron apresenta o sistema SOCRATES (“Systems for Organizing Content Review to and Teach Educational Subjects”) [120]. Este sistema instaura entre o aluno e o computador ao qual ele estaria ligado, um diálogo do tipo Socrático, com perguntas e respostas, que poderia conduzir o aluno à consulta na biblioteca da universidade, caso fosse necessário chamar um professor se o conhecimento apresentado trouxesse alguma dificuldade.

Em 1960, a Universidade de Illinois, coloca outro projeto de pesquisa e desenvolvimento no domínio do Ensino assistido por computador: o sistema PLATO (“Programmed Logic for Automated Teaching Operations”), dirigido por Donald Bitzer e desenvolvido no CERL (“Compute Education Research Laboratory”) [193]. Em 1961, são instalados numa sala do campus desta universidade, 25 terminais ligados ao computador ILIAC I. Acoplado ao ILIAC I existia um dispositivo eletromecânico complexo que permitia a visualização de eslaides. Em 1964, o ILIAC foi substituído

por um computador mais poderoso, o Control Data 1604, que permitia a difusão dos tutoriais e suas simulações, cujos diagramas eram visualizados primeiramente em telas colocadas em pontos da universidade e posteriormente em terminais táteis de vídeo nos quais os alunos controlavam o computador. O PLATO espalhou-se nos Estados Unidos e na Europa sobre a bandeira comercial da Control Data. O tempo de acesso e as respostas eram relativamente lentos e o sistema era operacionalmente muito caro. O PLATO gerou um considerável bancos de programas que iam desde exercícios repetitivos até tutoriais com execução de simulações mais sofisticadas, com ajuda da linguagem de autoria TUTOR [44]. Em 1982, diversos programas do sistema PLATO [120] foram adaptados pela TenCore para microcomputadores, incluindo a linguagem TUTOR, denominados então de EnBasic [45] conforme mostra a Figura 2.7.



Figura 2.7: EnBasic: adaptação dos programas do PLATO. Fonte: [45]

Em 1963, Patrick Suppes [207] do Instituto de Estudos Matemáticos da Universidade de Stanford, inicia um programa de pesquisa e desenvolvimento de Ensino assistido por computador, particularmente no domínio do Ensino de Matemática. As pesquisas se consolidam na criação de numerosos conjuntos de exercícios e tutoriais como resultados de suas experiências elaboradas sobre as etapas da aprendizagem. Os programas eram testados com os alunos e revisados antes da distribuição. Em colaboração com a IBM, o grupo de Stanford aperfeiçoou o sistema de autoria “Coursewriter”.

Em 1964, a IBM finaliza uma tela de raios catódicos especialmente desenvolvida para Ensino assistido por computador. Assim, em 1966, com a participação do grupo de Suppes, é proposto o IBM 1500, o primeiro computador destinado ao Ensino assistido por computador.

Em 1967, Suppes apud Barr [30] desenvolvem o EXCHECK para Ensino de Lógica e Teoria dos Conjuntos. Este programa reagia às respostas dos alunos com mensagens instrucionais. Pode-se dizer que este programa foi o precursor da mode-

lagem qualitativa pois representava a matéria através de demonstrações e exemplos. O trabalhos de Suppes [207] e seus colaboradores foram comercializados posteriormente pelo “The Computer Curriculum Corporation”. No início dos anos 80, estes trabalhos também foram transportados para microcomputadores.

Ainda em 1967, dando continuidade aos esforços de Bitzer (PLATO) e de Suppes, um comitê científico apresentou, para o presidente dos Estados Unidos recomendações quanto ao uso de computadores na alfabetização de crianças nas escolas americanas. Inicia-se, assim, a formação de um sindicato constituído de um grupo de escolas de Minnesota, denominado TIES (“Total Information for Education Systems”), que incentivava o uso do computador na escola.

Baseando-se nos sistemas TUTOR [120] e Coursewriter, John Starkweather, do Centro Médico da Universidade da Califórnia, finaliza, em 1969, outro sistema de autoria, denominado PILOT, que atinge um certo sucesso comercial. Assim, em 1973, começam a ser padronizadas diversas versões baseadas no PILOT [44]. Uma dessas versões foi posteriormente desenvolvida para computadores Apple II, Commodore 64 e PC conforme mostra a Figura 2.8.



Figura 2.8: TUTOR: versões para computadores Apple II, Commodore 64 e PC

De acordo com Bordeleau [44] a primeira aplicação educativa do hipertexto foi feita pelo Professor Vam Dam da Universidade Brown, em 1970 nos EUA. Para isso, foi utilizado um sistema hipertexto distribuído em terminais para auxiliar o Ensino de poesia inglesa. O sistema continha a informação biográfica dos poetas estudados e as datas referentes ao estilo poético de cada um. Estas informações eram colocadas em relação aos comentários do professor em cada tópico estudado. Neste mesmo ano, de 8 a 10 de julho, acontece em Bloomfield Hill, Michigan, a “National Conference on Computer Application to Learning”.

Em 1971 os pesquisadores americanos do “National Education Committee” [45] fazem uma forte crítica ao uso da educação com recursos audiovisuais em um relató-

rio que proclama como solução a tecnologia educacional. A intenção era melhorar o Ensino e a aprendizagem de acordo com uma abordagem sistemática que ligava seres humanos aos meios tecnológicos, principalmente o computador.

Em 1978, o projeto PLATO [120] está em sua Quinta geração. Os terminais da Control Data tornam-se uma realidade para microcomputadores graças ao microprocessador de 8 bits 8080 da Intel, que executam operações autônomas, o que anteriormente necessitava dos recursos de um grande computador.

No início dos anos 80, a BBC de Londres promoveu um concurso para a escolha de qual computador seria utilizado pela “Open University” no Ensino à distância. A escolha foi dada à uma máquina baseada em 8 bits, embora, naquela época, já existissem máquinas dotadas com processador de 32 bits.

Em 1985, o governo da França fez uma compra enorme de computadores para distribuição nas escolas em todo território francês. Nas grandes cidades francesas as condições não eram as mesmas que no interior. Não havia professores preparados para o manuseio destas máquinas o que fez com que grande parte dos computadores não fossem utilizados levando o plano ao fracasso, exceto para a companhia que vendeu os computadores. Na mesma época, a Bélgica, em iniciativa semelhante, incentivou o uso do computador nas escolas. O governo belga liberou verbas para as universidades criarem centros de ensino de computação. Estes centros possuíam cerca de 4 a 5 professores dedicados ao ensino com computador cuja principal tarefa deles era a confecção de cursos introdutórios de computação para serem ministrados a professores do ensino secundário e primário. Além dos cursos, havia a demonstração de programas para ensino. Após a obtenção dos conhecimentos básicos em computação, os professores do ensino secundário e primário poderiam solicitar ao governo, através de projetos, os computadores para uso nas escolas. Este procedimento funciona até os dias atuais e nota-se que é algo bem pensado e equilibrado pois, ao invés de se colocarem computadores nas escolas sem a preparação prévia dos professores, estes são preparados pelas universidades e, motivados, requerem equipamentos de acordo com suas necessidades.

Ainda num contexto educacional atualizado pode-se pensar que o computador possui diferentes papéis, desde o apoio aos professores e estudantes até a geração de comunidades virtuais[2], [41], [66], [83], [114], [122], [179], [183], [215], utilizando ferramentas de IA [133], [178], [201]. Embora muitas destas aplicações ainda se limitem, no Brasil, aos centros tecnológicos, pode-se prever, em poucos anos, sua disseminação aos “cidadãos comuns”, como foi o caso do acesso à Internet. Em 1989, a então BITNET, de uso exclusivo de universidades e centros de pesquisas no Brasil fez sua primeira comunicação com o exterior. No final de 1994, tornou-se acessível através de provedores comerciais.

2.7 Inteligência Artificial no Ensino

As primeiras tentativas do que pode ser denominado de “Ensino inteligente auxiliado por computador” estão nos trabalhos com Sistemas Especialistas (SE) [30], [57]. Nestes sistemas, tentava-se imitar um professor, especialista em um domínio, através da modelagem de Ensino voltada para as características dos alunos [94], [160]. Tentava-se, assim, resolver problemas causadas pela limitação dos sistemas de Ensino, auxiliado por computador clássico, que eram incapazes de:

- Conversar com o aluno em linguagem natural;
- Entender a disciplina de forma a responder perguntas inesperadas;
- Decidir o que deve ser ensinado nos passos seguintes;
- Antecipar, diagnosticar e entender os erros dos alunos e suas falsas concepções;
- melhorar e modificar estratégias de Ensino progressivamente ou na aprendizagem de novos conceitos.

Em 1970, Carbonell [51] desenvolveu o SCHOLAR para Ensino de geografia da América do Sul, que utilizava a modelagem do estudante. A representação do conhecimento estava contida numa base de dados organizada em redes semânticas. Por exemplo, todos os dados relativos à cidade do Rio de Janeiro eram reagrupados em um nó da rede que permitia formular algumas regras em níveis superiores da rede e que podiam ser compartilhadas com outros nós inferiores. Além disso, um conjunto de regras permitia o computador gerar conhecimento quando confrontava-se com uma pergunta particular [44].

Em 1973, foi desenvolvido o INTEGRATION tendo como sua principal característica a integração do conhecimento do domínio com o modelo do estudante, de maneira a direcionar a seqüência de Ensino [34].

Em 1974, é lançado o SOPHIE (com as versões I, II, III) que ensinava, através de tentativas e erros e uma base de conhecimento contendo regras e procedimentos, a detecção de defeitos em circuitos elétricos. Talvez seu maior mérito seja motivar o estudo da física qualitativa e pode ser considerado o pioneiro no uso da simulação [34].

Em 1973, Shortliffe [198], [199] lança o MYCIN, utilizado para auxílio no diagnóstico de doenças infecciosas do sangue, sugerindo o tratamento. Neste sistema existe a separação entre a base de conhecimento e o modo de manipulação do conhecimento [30].

O WHY , implementado em 1977 por Stevens e Collins [206] usava a técnica socrática, na qual as perguntas são formuladas para provocar a reflexão do aprendiz sobre o assunto.

A concepção do BUGGY , em 1978, fazia um modelo psicológico do aluno para detectar erros conceituais em cálculos aritméticos. Possuía mecanismos de aprendizagem de como os alunos resolviam os problemas [120].

MUMATH é um programa de solução de problemas de matemática simbólica tais como cálculo matricial, cálculo diferencial e integral. O programa não apresenta conjunto de lições, mas oferece, como LOGO, um ambiente experimental, com uso de programa capaz de tratar símbolos, o que colabora no Ensino da matemática [120].

O GUIDON , construído por Clancey [57] em 1979, inicialmente usando a base de conhecimentos do MYCIN pode, em princípio, ser utilizado em outros domínios de conhecimento. O sistema caracteriza-se pelo uso de graus de pertinência da mesma forma usada em conjuntos nebulosos e permite um diálogo da mesma maneira que o MYCIN [58] na resolução de um problema em particular. O estudante poderia comparar seu progresso ao de um especialista porque o sistema era capaz de explicar seu próprio raciocínio. Os trabalhos de Clancey [58], [59], [60] e [61] abordam os aspectos construtivos de sistemas de tutoria baseados em conhecimentos.

O PROFCOMP [24], no início dos anos 80, introduziu simulação e conceitos de hipertexto no Ensino de circuitos [162]. O SIMED [168] integrou conceitos de hipertexto [163], [164], [170], simulação e IA no Ensino da medicina [39], [131],[213]. Detalhes sobre estes sistemas podem ser obtidos em Pagano [165], [166], [167] e [169].

O MENO-TUTOR , segundo Ramírez [182], de 1984, não possui nenhum domínio específico; preocupa-se basicamente com o módulo de Ensino contendo regras pedagógicas. Este módulo é descrito como um conjunto de unidades de decisão organizadas em três níveis de planificações que vão refinando sucessivamente as ações do tutor.

Segundo Almeida [5] o quadro 2.1 mostra outros esforços proeminentes historicamente no desenvolvimento de programas de Ensino com IA.

Em 1988 surgiu o APLUSIX, para Ensino de manipulação algébrica, seguindo o modelo clássico de ICAI, com um SE baseado em regras, um módulo pedagógico, um modelo de estudante e uma interface ergonômica [182].

WUMPUS e WEST [120] são dois jogos aos quais se incorporou um módulo para orientar jogadas e ensinar a jogar. Os jogos podem funcionar de modo totalmente independente da parte de Ensino.

ITS	Desenvolvedor	Ano	Domínio	palavra-chave
DEBUGGY	Burton	1982	aritmética	diagnósticos “off-line”
STEAMER	Hollan	1983	projeto de navios	simulação, modelos mentais
LMS	Sleeman	1984	álgebra	regras
MENO	Woolf	1984	metereologia	regras pedagógicas
PROUST	Johnson	1984	programação	diagnose
ACTP	Anderson	1984	tutor LISP	modelagem cognitiva
SIERRA	Vanlehn	1987	aritmética	predição de erros
SHERLOCK	Lesgold	1991	eletrônica de AF	aprendizagem cognitiva

Tabela 2.1: Alguns ITS históricos - adaptado de [120]

Uma alternativa para o desenvolvimento de sistemas que rompiam com o Behaviorismo foi a concepção das linguagens LOGO e do SMALLTALK [43].

O LOGO é uma extensão da linguagem LISP desenvolvida por Seymour Papert [173]. Esses sistemas partem do princípio de que é necessário encorajar o estudante a programar a máquina e não o contrário. A primeira versão do LOGO foi introduzida, em 1966, no MIT, como uma influência dos estudos de Papert sobre os trabalhos de Jean Piaget [176], [177]. Papert introduziu a versão Construtivista [69] do treinamento computadorizado, na qual o estudante poderia construir seu próprio conhecimento através da exploração, enquanto trabalhava com objetos de conhecimento apropriados. Em 1970, nasce então a metáfora da tartaruga gráfica, um pequeno cursor que se move na tela e desenha algumas formas geométricas de acordo com as instruções dadas ao computador por uma criança que programa em LOGO [68].

Em 1979, Adele Goldberg, da Xerox de Palo Alto, Califórnia experimenta o ambiente SMALLTALK com crianças [43]. Este sistema propõe que, os usuários adultos ou crianças, podem modelar as ferramentas que lhe são necessárias, particularmente na Educação.

Uma criança que trabalha com LOGO utiliza procedimentos e listas e entende a importância dos estágios do programa e o princípio da recursividade; aquelas que trabalham com SMALLTALK utilizam a idéia de objetos e classes. Segundo Goldberg apud Bordeleau [44], o SMALLTALK é provavelmente mais um protótipo do que um paradigma de programação acabado e teve uma grande influência nos anos 90 como um paradigma orientado a objetos que auxilia o entendimento do ambiente computacional.

Existem vários níveis de possibilidade do uso da IA na concepção de sistemas de Ensino. A cada um desses níveis correspondem técnicas particulares da IA, e necessidades freqüentemente diferentes, promovendo a interação dos diversos campos do conhecimento humano [72]. Pela interação com outros assuntos, o uso da IA no Ensino torna-se não somente uma forma mais adequada de transmissão de conhecimentos, mas um estímulo a novas pesquisas.

Muitos estudos recentes reforçam a atenção dos pesquisadores na promoção do diálogo humano com o computador. Um modo de compreender um assunto é ter de explicá-lo. As técnicas de IA necessitam então de uma explicação do conhecimento. E é assim que se pode esperar dos estudos para aplicações ao Ensino uma melhor compreensão de todo o processo de aprendizado. Uma forma mais interessante e recente de classificar tais sistemas é utilizar conceitos de IA distribuída com a modelagem de agentes [217]. Nestes sistemas tem-se os modelos de estudantes, de especialistas e de interface com usuário apropriados ao processo de Ensino-aprendizado [209].

A parte da IA, denominada *Inteligência Artificial Construtivista*, trata das estratégias pedagógicas a serem desenvolvidas por estes agentes de programas. Vários trabalhos, utilizando orientação a objetos, podem ser encontrados nos exemplares da Revista Brasileira de Informática na Educação ³. Alguns destes sistemas [126] têm alcançado relativo sucesso e podem ser obtidos gratuitamente no Laboratório de Software Educacional (EDUGRAF) do Departamento de Informática e de Estatística (INE) da Universidade Federal de Santa Catarina (UFSC).

A utilização da IA como auxílio de ferramentas de Engenharia de Programas, Telecomunicações e Paradigmas Educacionais pode servir para o desenvolvimento de ambientes de aprendizagem distribuídos e cooperativos.

³A Revista Brasileira de Informática na Educação (RBIE) é uma publicação da Sociedade Brasileira de Computação (SBC), editada pelo Departamento de Informática e de Estatística (INE) da Universidade Federal de Santa Catarina (UFSC).

2.8 Aprendendo no Ciberspaço

A denominação “cyberspace”, que passou a ser utilizada para identificar os ambientes criados em computador, origina-se do livro “*Neuromancer*” de William Gibson [81].

O Ciberspaço ⁴ pode oferecer vários exemplos de cenários de aprendizagem. Para as simulações em ambiente virtual pode-se utilizar técnicas de IA distribuída e RV. A RV consiste numa confluência de tecnologias que, além da Computação Gráfica Interativa, inclui a holografia, com a utilização de lasers para a criação de imagens tridimensionais, visores de cristal líquido, televisões de alta definição, multimídia e outros.

Um humano, dentro de um mundo virtual, é um “Avatar” . Dentro desses mundos, no entanto, podem existir habitantes inteligentes, os “Agentes Cognitivos”. Os usuários interagem com realidades artificiais, como se estivessem no mundo real, usando a fala e os movimentos como mecanismos de interação. Um conjunto de imagens ajudam ao usuário na interpretação da informação apresentada. Estas imagens se comportam de uma forma similar a do objeto ou entidade que estão representando. Os usuários interagem com essas realidades como se fossem habitantes do mesmo mundo virtual.

A Figura 2.9 mostra um dispositivo de RV com as entradas e saídas motoras mediadas por computador.



Figura 2.9: Entradas e Saídas Motoras mediadas pelo computador.

⁴Também denominado de *Espaço Cibernético*.

No começo dos anos 60, desenvolveu-se o simulador “Sensorama Arcade” que utilizava estímulos de visão, audição, movimento e cheiro para a imersão do participante. Em 1965, Sutherland projetou um “Head Mounted Display” (HMD) que muda a visão apresentada em decorrência dos movimentos dos usuários. Em 1985, o centro de pesquisa AMES da NASA começou a desenvolver um equipamento de RV, a baixo custo. A Figura 2.10 mostra o “Virtual Interactive Environment Workstation (View)”, consistindo em um capacete que oferece uma visão estereoscópica tridimensional e 360 graus de visibilidade. Este equipamento foi desenvolvido pela NASA, em 1987, para que um astronauta pudesse comandar um robô. As informações eram enviadas por um sensor Polhemus (nome do fabricante) que era capaz de seguir a direção do olhar [81].



Figura 2.10: “N.A.C. Eye”[81]

Dentro das possibilidades de interagir com o meio ambiente, pode-se conceber a RV, enquanto produto de consumo, como mecanismos pelos quais são excitados diferentes receptores neuronais, de forma a provocar determinado tipo de reações. Na prática, identificam-se os seguintes conjuntos de aplicações que podem surgir sozinhos ou se comporem:

- Simuladores;
- Vida Artificial;
- Redes de Computadores;
- Telepresença;
- Interfaces inteligentes;
- Ambientes virtuais.

As primeiras aplicações foram os simuladores nos quais, dentro de um computador, eram construídas toda uma “realidade”. Esses simuladores serviam para

reproduzir sistemas de forma a representá-los não mais pela forma estática de modelos matemáticos mas sim, enquanto processos dinâmicos. Uma teoria pode ser simulada, e sua qualidade verificada medindo-se a distância entre os resultados reais obtidos e os verificados dentro do espaço cibernético artificial, que é constituído pela tela do computador e toda circuitaria e programas a ela associados.

Outra aplicação típica em RV consiste na criação de Autômatos Celulares - criaturas que devem obedecer um conjunto de regras estabelecidas dentro de um mundo para elas criado dentro de um computador. As criaturas mais adaptadas sobrevivem e, ao realizar esse feito, procriam, transmitindo seus genes às novas gerações que se comportarão de uma forma nova, herdada geneticamente de seus pais, sujeita à mutações. A esta parte da computação denomina-se “Computação Evolucionária” cujos principais paradigmas são: estratégias evolucionárias, algoritmos genéticos e programação evolucionária. Estes paradigmas são trabalhos computacionais inspirados biologicamente na Teoria da Evolução das Espécies. Uma explicação bastante simples sobre a Computação Evolucionária pode ser encontrada em Barreto [37].

As redes de computadores, ligadas a aparelhos ou mecanismos que permitem interagir com qualquer realidade artificial construída, multiplicam-se, emaranhando-se em milhares de dispositivos e milhões de usuários. Exemplos são a TV iterativa, as Redes de Dados como a INTERNET , INTERNET 2 , etc.

As redes sem fio, podem reduzir bastante os custos de implementação e configuração física de sistemas dedicados ao Ensino. Segundo Fialho [81]:

“... Estamos construindo um sistema de informações que, inspirado na Rainha Mab de Shakespeare, é denominado de “World Wide Web” (teia de aranha do tamanho do mundo). A Internet com seus milhões de usuários de toda a parte do mundo parece ser o meio ideal para implementar esse novo mundo de subjetividade no qual, cada vez mais, estamos imersos”.

A “World Wide Web” tem sido denominada “quarta mídia” e promete para o Séc. XXI tomar seu lugar ao lado da imprensa, rádio e televisão, interagindo com estas. Os portais web crescem monstruosamente na rede de computadores mundial, a cada dia. Em servidores WWW, os hipertextos, além de relacionar informações, permitem a navegação entre vários servidores espalhados pela rede, possibilitando o acesso a uma enorme quantidade de dados. Os hipertextos podem conter gráficos, imagens, arquivos de áudio ou ainda animações, e, neste caso, são denominados de Hiperídia .

Com o desenvolvimento de tecnologias interativas que possibilitam contato em tempo real entre locais espalhados geograficamente, começam a surgir as chamadas

“classes virtuais”. Dentre as principais características destas novas salas de aula, pode-se citar a possibilidade de contato com um largo espectro de colaboração dos aprendizes, numa quantidade bastante superior ao que podem encontrar em sua própria região, além de permitir o acesso a um quadro bastante extenso de professores e mentores, numa dimensão impossível para uma única instituição educacional local.

A utilização da rede mundial visa promover uma maior interação aluno-professor e aluno-aluno, como um espaço de troca e produção coletiva de conhecimento e informação, fora dos horários de aula. Essa interação acontece através do site WWW, disponível para os alunos e demais envolvidos no curso, onde o aluno encontra um conjunto de ferramentas multimídia, desenvolvido especialmente para os cursos oferecidos. No local estão as ferramentas necessárias para o aluno comunicar-se com seus professores ou colegas, comentar as aulas, discutir temas relacionados às disciplinas em andamento, enviar sua produção ao professor e acessar ementas de disciplinas, bibliografias de referência, artigos e outras informações importantes para um bom desempenho no curso.

A telepresença permite executar atividades em domínios geograficamente ou fisicamente distintos através da robótica e dos sistemas de comunicação [81]. Um exemplo são os “waldos” que multiplicam forças ou ampliam a capacidade dos sentidos humanos, podendo até traduzir, para o mundo da miniatura, operações que os humanos seriam incapazes de executar dada a delicadeza e precisão exigidas. A expressão “waldo” surgiu, pela primeira vez, em um livro de ficção científica de Robert A. Heinlein [81]. Sendo profundamente inteligente, porém limitado fisicamente, Waldo construiu máquinas capazes de “interficiar” suas relações com o mundo transformando-se de pessoa deficiente em verdadeiro super-homem. A NASA dispõe de uma aplicação dentro dessa linha que se traduz pelo conceito de telepresença robótica. Um astronauta pode realizar consertos na parte externa de uma estação espacial por meios de robôs controlados no interior da nave.

Muitos afirmam que os ambientes virtuais “prometem” revolucionar o Ensino. Segundo Fialho [81], oficinas que podem ser acessadas por qualquer dos meios anteriormente citados podem fornecer os estímulos capazes de despertar nos alunos a paixão pela aprendizagem. O professor teria um duplo papel, o de criador dessas oficinas de estímulos e o de incentivador dessas paixões.

2.9 Considerações

Pode-se discutir o mérito do Ensino por máquinas, mas é incontestável que elas vêm auxiliando o processo de Ensino-aprendizagem ao longo dos anos. E a cada dia são feitos esforços para que elas possam ser ferramentas úteis na construção dos

conhecimentos humanos.

Um grande passo neste sentido é tornar as máquinas capazes de aprender. A evolução do computador e das estratégias educacionais, com certeza, colaboram para o desenvolvimento dos ambientes de Ensino. Entretanto, não devem ser esquecidos os papéis do professor e do aluno. O papel fundamental do professor é utilizar a máquina como uma ferramenta para ensinar e aprender. O papel do aluno é “aprender a aprender” ampliando suas capacidades intelectuais, para também ensinar.

Os primeiros sistemas de Ensino computadorizados foram muito influenciados pelas teorias comportamentalistas. Atualmente, a adoção da filosofia Construtivista é amplamente divulgada para concepção de sistemas computadorizados em diversos domínios do conhecimento. Também os sistemas, utilizando IA, revisaram seus paradigmas educacionais e daí surgiu a IA Construtivista. Entretanto, não basta a adoção deste ou daquele paradigma educacional. A construção correta computacional de programas é a base da EP. Assim, para que um sistema de Ensino seja eficaz, aliado aos paradigmas educacionais, deve ser focalizado como é construído corretamente do ponto de vista computacional.

A IA no Ensino⁵, aliada às redes de computadores, são poderosos recursos da Computação para melhoria da qualidade dos programas de Ensino. Entretanto, deve-se lembrar que a especificação correta dos sistemas são altamente desejáveis para seu funcionamento correto e eficácia no processo de Ensino-aprendizagem.

A abordagem de Engenharia de Programas em IA iniciou-se na Europa, no final dos anos 80. Pesquisadores envolvidos na construção de SE utilizando IAS (que envolviam milhares de regras), começaram a utilizar técnicas de Engenharia de Programas para facilitar a construção destes sistemas. Foi assim que nasceu o projeto KADS (KADS I e KADS II) [108] financiado pela CEE (“Comission Économique Européenne”) em vários países. KADS significava, quando da adoção da sigla, “Knowledge Acquisition and Development Method”. O KADS foi um dos primeiros trabalhos relativos à confecção de SE com metodologia de Engenharia de Programas. Foi um primeiro esforço de se ter uma definição formal de Ciclo de Vida permitindo que várias pessoas colaborem num projeto de um SE, maior, que possa viver mais tempo. É também interessante considerar o Ciclo de Vida de programas educacionais desde que este seja um programa educacional integrado. Se o programa educacional é utilizado para Ensinar um pequeno módulo de uma disciplina não há necessidade de todo esse esforço mas, se o desejo é integrar a disciplina como um todo, em termos de ferramentas computacionais, para seu apoio, (por exemplo, via laboratório virtual) o esforço vai envolver mais de uma pessoa e o modo de

⁵Uma leitura mais detalhada sobre as “Perspectivas da Inteligência Artificial no Ensino” pode ser encontrada no trabalho de Almeida [5].

passar a experiência de uma para outra pessoa será através de documentos formalizados. Neste caso, deve existir uma normalização dos documentos que devem ser produzidos.

A introdução de *Engenharia de Programas Educacionais* não diminui a necessidade de aplicação de paradigmas educacionais, apenas complementa, do ponto de vista educacional, a concepção destes ambientes. As Máquinas para Ensinar, denominadas computadores, tornar-se-ão, então, ferramentas mais eficientes quando usadas na Educação.

Esta tese enfatiza que algo semelhante ao KADS deva ocorrer em relação aos programas educacionais. Seria muito interessante examinar se as particularidades detectadas no KADS para a construção de SE aplicam-se (ou seja, mantêm-se ou não) no caso de Ensino com computador, pois as características do Ensino estão muito próximas da IA.

Uma definição teórica de um modelo Hipermídia, fundamentado na Teoria de Autômatos, poderá aumentar o potencial dos sistemas computacionais de Ensino quando considerados seus aspectos dinâmicos. A Teoria das Categorias visa simplificar o processo de especificação através da abstração de conceitos. Os conceitos de Teoria das Categorias, Hipermídia como Autômato são abordados nos próximos capítulos desta tese.

Capítulo 3

Cognição e Computação

*“I propose to consider the question:
‘Can machines think?’
This should begin with definitions of
the meaning of the terms ‘machine’ and ‘think’”.*

Alan Turing, “Computing Machinery and Intelligence”, 1950.

Neste capítulo são apresentados os paradigmas da Ciência Cognitiva e um estudo comparativo entre o aprendizado animal e o aprendizado de máquina (“Machine Learning”). É apresentada uma taxonomia do aprendizado de máquina enfatizando as estratégias de aprendizado e a representação do conhecimento. Considerando que o aprendizado é a base da inteligência, com a finalidade de mostrar a potencialidade dos métodos formais, é oferecida uma resposta formal à clássica pergunta: “A Inteligência Humana é uma Função Computável”?

3.1 Ciência Cognitiva

Segundo Rapaport [184] “... *A Ciência Cognitiva é o estudo interdisciplinar da cognição.*”. A cognição inclui os estados mentais e os processos como pensamento, raciocínio, memorização, geração e compreensão de linguagem, percepção visual, aprendizado, consciência, emoções, etc.

Alguns cientistas cognitivos limitam seus estudos à cognição humana, embora a

maioria considera a cognição independente de suas implementações em seres humanos ou computadores. Outros, estudam a cognição independentemente dos agentes cognitivos ambientais e outros a estudam dentro do contexto da sociedade ou cultura.

A Ciência Cognitiva ou Cognitivismo, estuda a mente em contraste com o Behaviorismo. Este, por sua vez, estuda o comportamento humano em termos das correlações de estímulos e respostas, sem nenhuma menção aos estados mentais inobserváveis (incluindo construções mentais tais como símbolos, idéias, esquemas) ou processos mentais (tais como pensamento, planejamentos, etc.) que possam mediar estas correlações. Um behaviorista vê a mente como uma “caixa preta” que somente pode ser entendida em termos de comportamento entrada-saída. Os Cognitivistas buscam o entendimento da mente humana em termos dos estados e processos mentais, ou seja, em termos dos algoritmos que mediam a entrada e saída.

A Ciência Cognitiva teve suas origens em duas grandes linhas de pesquisa. A primeira foi o desenvolvimento da Lógica Simbólica até a sua aplicação por McCulloch e Pitts [144] em 1943 no comportamento das Redes Neurais. A segunda linha, é devida a Turing em sua análise computacional (1936) e sobre a possibilidade de máquinas pensantes (1950) conhecido como o Teste de Turing [120]. De acordo com Rapaport [184] o Cognitivismo entra em cena em 1956 seguido de várias teorias [195]. Os trabalhos de Simon e Newell [157], [158] originam o “Logic Theorist”, um dos programas pioneiros de IA [48], [134], [142]. Em 1979, surge o “Journal of Cognitive Science” e dois anos mais tarde ocorre o primeiro encontro anual da “Cognitive Science Society” [184]. Um dos maiores teóricos do Cognitivismo foi Jean Piaget que desenvolveu seus trabalhos no início dos anos 20. As idéias de Piaget [175], [176], [177], não causaram impacto nos EUA até nos anos 60, quando da fundação do “Harvard Center for Cognitive Studies” [100]. Jean Piaget nasceu em 1896 em Neuchâtel na Suíça. É conhecido como o fundador da Epistemologia Genética¹, que descreve o desenvolvimento intelectual em termos da maturidade e da experiência. Assim, ao processo de construção do conhecimento através do desenvolvimento intelectual, segundo Piaget, denominou-se *Construtivismo*.

“Cognitive theorists recognize that much learning involves associations established through contiguity and repetition. They also acknowledge the importance of reinforcement, although they stress its role in providing feedback about the correctness of responses over its role as a motivator. However, even while accepting such behavioristic concepts, cognitive theorists view learning as involving the acquisition or

¹No prefácio do livro “Embodiments of Mind” de McCulloch [143], Papert sugere que a Epistemologia Genética seja inspirada nas idéias de McCulloch, filósofo que apresentou o primeiro modelo de neurônio.

reorganization of the cognitive structures through which humans process and store information” [100].

O *Cognitivismo* relaciona a natureza do aprendizado com as estruturas cognitivas que são esquemas (pensamentos) que um organismo possui em um dado momento e com os quais interage com o seu ambiente físico. Uma estrutura cognitiva resulta da maturação biológica e da experiência acumulada. A estrutura cognitiva não só é afetada pela experiência mas também determina o que pode ser experimentado [212]. Se um evento físico não pode ser assimilado parcialmente na estrutura cognitiva de um organismo então este evento físico não constitui um estímulo biológico.

Dentre os vários conceitos importantes da Teoria Cognitivista no aprendizado humano têm-se [100]:

- *Esquema*: uma estrutura interna de conhecimento. Uma nova informação que é comparada com uma existente é denominada esquema. Cada esquema pode ser combinado, estendido ou alterado para acomodar a nova informação.
- *Modelo dos três estágios para processamento da informação*: inicialmente a entrada entra num registrador sensorial, depois é processada em uma memória de curta duração e então é transferida a uma memória de longa duração para armazenamento e recuperação.
- *Registrador sensorial*: recebe a entrada de sensações, que duram de um a quatro segundos e então desaparece por decaimento ou substituição. Grande parte da informação nunca alcança a memória de curta duração mas toda informação é monitorada em algum nível e é ativada se necessário.
- *Memória de curta duração (“Short-Term Memory (STM)”)*: a entrada sensorial que for importante ou interessante é transferida do registrador sensorial para a STM. A memória pode reter a informação por mais de 20 segundos. A capacidade da STM pode aumentar se a informação for colocada em partes significativas.
- *Memória de longa duração e armazenamento (“Long-Term Memory and Storage” (LTM))*: armazena a informação da STM para uso a longo prazo. Possui capacidade ilimitada. Muitas informações são “forçadas” na LTM através do aprendizado por decorar. A LTM, através de níveis de profundidade do processamento, gera ligações entre velhas e novas informações.
- *Efeitos significantes*: a informação significativa é mais fácil de se aprender e de se lembrar. Se um aprendiz liga uma informação relativamente menos significativa com uma informação de um esquema anterior, esta será facilmente retida.

- *Efeitos de posição consecutiva*: é mais fácil lembrar itens do início ou do fim de uma lista do que os do meio da lista, a menos que estes sejam distintivamente diferentes.
- *Efeitos práticos*: praticar ou refazer aumentam a retenção de conceitos.
- *Efeitos de transferência*: são efeitos de aprendizado anterior no aprendizado de novas tarefas ou conceitos.
- *Efeitos de interferência*: ocorrem quando o aprendizado anterior interfere no aprendizado de novos conceitos.
- *Efeitos de organização*: quando o aprendiz organiza a entrada como, por exemplo, memorização de uma lista de supermercado.
- *Níveis de efeitos de processamento*: as palavras podem ser processadas no baixo nível sensorial pela análise de suas características físicas e ao alto nível pela análise semântica de seu significado.
- *Efeitos dependentes do estado*: se o aprendizado está dentro de um certo contexto será mais fácil lembrar dentro daquele contexto do que em um novo contexto.
- *Efeitos mnemônicos*: mnemotécnicas são estratégias usadas pelos aprendizes para organizar uma entrada relativamente sem sentido em imagens mais significativas ou contextos semânticos.
- *Efeitos de Esquemas*: se a informação não se ajusta ao esquema de uma pessoa, pode ser mais difícil para ela conceber ou se lembrar da informação e também isto pode ser afetado por esquemas anteriores.

3.2 Psicologia e Filosofia Computacional

O objetivo da Psicologia Computacional é entender o comportamento humano, dito inteligente, através da criação de programas de computadores que comportam-se da mesma maneira que as pessoas. Para alcançar este objetivo, é importante que o algoritmo expresso pelo programa seja o mesmo que as pessoas realmente usam e que as estruturas de dados utilizadas sejam as mesmas utilizadas na mente humana. O programa deve executar rapidamente o que as pessoas fazem rapidamente e executar lentamente o que as pessoas têm mais dificuldade de fazer. Também deve tender a cometer erros da mesma forma que as pessoas tendem a cometê-los. Se o programa

for colocado em situações experimentais, nas quais os humanos estariam sujeitos, os resultados deveriam estar dentro de uma faixa da variabilidade humana.

A Filosofia Computacional objetiva formar um entendimento computacional do comportamento, em nível inteligente humano, sem estar restrito a algoritmos e estruturas de dados que os humanos realmente utilizam (ou deveriam usar). Pode-se dizer que o “entendimento computacional” é um modelo que expressa um procedimento, no mínimo, implementável (se realmente não for implementável) no computador. O “comportamento em nível inteligente humano” é o comportamento, que quando desenvolvido por uma pessoa, é parte de um comportamento cognitivo humano dito inteligente. É aceitável, naturalmente, que o modelo implementado possa desenvolver algumas tarefas melhor do que qualquer pessoa.

As fronteiras se sobrepõem quando se trata de saber como programar computadores, especialmente para desenvolver tarefas que, embora não se saiba como programá-las, as pessoas desenvolvem. Assim, tanto a Psicologia como a Filosofia Computacional são subáreas da IA divididas de acordo com seus objetivos. A IA é considerada como um campo da Ciência da Computação.

Existem diversas disciplinas externas à Ciência da Computação, que provocam fortes impactos na IA e vice-versa. A Psicologia Cognitiva é um campo da Psicologia que usa métodos experimentais para estudar o comportamento cognitivo humano. A Psicologia Computacional, obviamente, está fortemente relacionada com a Psicologia Cognitiva, diferindo, principalmente, no uso de modelos computacionais ao invés de experimentos humanos. Todavia, os pesquisadores da IA estão atentos às pesquisas da Psicologia Cognitiva, assim como a Psicologia Cognitiva observa os procedimentos cognitivos nas pesquisas em IA que, possivelmente podem ser encontrados em seres humanos.

A Ciência Cognitiva é, então, um campo interdisciplinar que estuda o comportamento cognitivo humano sobre a hipótese da cognição ser (ou pode ser modelada como) uma computação. A IA [30], [161], [187], [190], [196], [223] e Ciência Cognitiva [184], [193] se sobrepõem. Além disso, a Ciência Cognitiva possui pesquisadores da Psicologia Cognitiva, da Lingüística, da Filosofia, da Antropologia, da Neurociências, entre outras.

3.3 Cognição e Computação

A noção que os estados e processos mentais intervêm nos estímulos-respostas às vezes são denominados de “metáfora computacional”. A mente está para o cérebro assim como o programa está para a circuitaria; os estados e processos mentais são como programas de computadores implementados (no caso em humanos) em estados e

processos cerebrais. Alguns cientistas, como Pylyshyn [180], [181] chegam a afirmar que: “a cognição é um tipo de computação”.

Assim, de acordo com a visão computacional da Ciência Cognitiva, pode-se afirmar que:

1. Existem estados e processos mentais intervindo entre os estímulos de entrada e a respostas da saída;
2. Estes estados e processos mentais são algoritmos (no sentido literal da palavra) ou são como algoritmos (no sentido metafórico);
3. Em contraste com o Behaviorismo, os estados e processos mentais são capazes de serem estudados cientificamente (mesmo que não sejam diretamente observados).

Neste sentido, uma questão filosófica é bastante discutida [85], [180], [181]. Se os estados e processos mentais podem ser expressos como algoritmos, eles são capazes de serem implementados em computadores não humanos. Estariam estes computadores executando tais algoritmos meramente estimulando estados e processos mentais ou estariam realmente pensando?

Alguns cognitivistas [85] discordam dessa denominada visão fortemente computacional da mente; e a eles são oferecidas respostas para assegurar uma espécie de teste de consistência desta teoria, ou seja, os cognitivistas favoráveis a esta visão oferecem respostas como “a visão computacional da mente é apenas um modelo de trabalho cujo comportamento pode ser comparado com o desempenho humano...”. Entretanto, a maioria concorda que o comportamento cognitivo humano pode ser simulado por um programa de computador e isto não é incoerente com esta teoria.

3.4 Paradigmas da Ciência Cognitiva

3.4.1 Ciência Cognitiva Computacional Simbólica

O Cognitívismo Simbólico Computacional é também denominado de “Physical Symbol System Hypothesis (PSSH)” ou “Representational Theory of the Mind (RTM)”. O PSSH, devido a Newell e Simon [157] [158], oferece a solução do problema de “como é possível para a mente existir em um universo físico”. A mente existiria fisicamente implementada como um “sistema simbólico”. Neste caso, um sistema simbólico é qualquer procedimento efetivamente computável, ou seja, uma máquina universal que pela tese de Church segundo Rapaport [184], poderia ser a Máquina

de Turing, uma função recursiva, um computador digital, etc. Um sistema físico simbólico é a implementação física de tal sistema simbólico.

De acordo com o PSSH [157]; um sistema físico é capaz de exibir comportamento inteligente (onde a inteligência é definida em termos de inteligência humana) se, e somente se, este sistema for um sistema físico simbólico.

Isto é uma hipótese empírica cujas evidências derivam-se dos trabalhos na IA simbólica. Newell [157] afirma que os *sistemas físicos inteligentes* são *sistemas físicos simbólicos* porque a inteligência requer *representações* de uma grande variedade de metas e estados e tais representações flexíveis necessitam de símbolos. Isto é, as representações devem ser simbólicas.

Nas linhas do PSHH e da RTM, quando um sistema físico (computador ou humano) executa um algoritmo “cognitivo”, as representações são trazidas “para a vida” como quem fala e se comporta de acordo com as regras de um sistema simbólico - o sistema torna-se dinâmico ao invés de ser estático. Se a cognição é baseada em representações e em regras (isto é, se o comportamento cognitivo consistir de transformações das representações de acordo com regras) então um computador que comporta-se de acordo com estas regras, aplicadas de maneira causal, para estas representações *está* se comportando cognitivamente e não meramente simulando o comportamento cognitivo. É claro que esta posição encontra seus opositores e dentre eles, Winograd [222].

3.4.2 Ciência Cognitiva Computacional Conexionista

O conexionismo (ou rede neural, ou processamento paralelo distribuído) é uma abordagem, tanto para a IA como para a Ciência Cognitiva Computacional, como sendo uma maneira pela qual um sistema parece comportar-se inteligentemente mesmo sem ser um sistema simbólico.

A maioria dos métodos conexionistas contém representações, mas estas estão distribuídas [63], [91], [95], [96] [127], ou seja, o conhecimento estaria distribuído nos pesos das conexões entre as unidades de processamento (neurônios). Assim, ao contrário de se ter uma inteligência “programada” no sistema através do uso de regras explícitas e representações, a inteligência “emerge” da organização dos nós e ligações entre os neurônios das redes neurais [35], [63], [91], [99], [121], [127], [148].

3.5 Aprendizado Animal × Aprendizado de Máquina

Uma característica óbvia do aprendizado humano é sua lentidão. Muitas vezes são necessárias décadas para que o ser humano aprenda alguma coisa. O aprendizado de máquina (“Machine Learning”) pode ser visto como um esforço para tornar o aprendizado humano mais rápido, mas este processo ainda é extremamente lento. Outra característica do aprendizado humano é que não podem ser feitas cópias. Num computador, uma vez que se tenha depurado o programa, compilado e rodado podem ser feitas muitas cópias. Em princípio, uma vez que um computador aprendeu algo, todos aprenderam; uma vez que se tenha um algoritmo para desenvolver determinada tarefa, este não precisa ser reinventado. Por exemplo, uma pessoa que sabe muito bem literatura não pode transferir, de modo direto, seu conhecimento e habilidade de interpretar textos para outra pessoa. Segundo Simon [200] *“uma pessoa sabe o que está em sua cabeça mas não sabe em que linguagem isto está codificado”*.

Um dos motivos da lentidão deste processo de aprendizado animal [132] seria a impossibilidade de cópias e transferência de programas. O computador faz o que a Psicologia denomina de “aprendizagem por tentativa”. Embora lento, o aprendizado humano está mais próximo do ótimo no desenvolvimento de diversas tarefas. É natural a procura pela construção de programas de computador que auxiliem na superação da ineficiência do aprendizado humano, mas existe a possibilidade de que muitos destes programas não possam ser construídos para desenvolver tarefas inerentemente humanas.

Uma definição satisfatória sobre o aprendizado de qualquer sistema (humano ou não) é que ele seja capaz de desenvolver, em face de mudanças, uma tarefa melhor do que desenvolveu anteriormente. Simon [200] afirma que:

“... Learning denotes changes in the system that are adaptive in the sense that they enable the system to do the same task or tasks drawn from the same population more efficiently and more effectively the next time...”

Para o aprendizado humano auxiliado pelo computador, a questão não é o ajuste do sistema humano para que tenha maiores capacidades. O interesse é que as pessoas adquiram a habilidade de desenvolver uma ampla faixa de tarefas auxiliadas por algo mais próximo delas - no caso, um computador “inteligente”. A Tabela 3.1 resume as comparações entre o Aprendizado Animal e o Aprendizado de Máquina.

Animal (Humano?)	Máquina (Computador)
lento (décadas)	tornar o aprendizado humano mais rápido
não podem ser feitas cópias	cópias após a depuração e compilação
desconhecimento da linguagem codificada	linguagem codificada
individual	universal
habilidade e interpretação	codificação
otimizado para tarefas diferentes	aprendizagem por tentativa, problemas não computáveis
Metáfora: mente	programa
Metáfora: cérebro	circuitaria
Inteligência Individual: indivíduo	Inteligência Individual: máquina
Inteligência Coletiva: espécie	Inteligência Coletiva: IA distribuída

Tabela 3.1: Aprendizado Animal \times Aprendizado de Máquina

Segundo Carbonell [52] “... *Learning is a many-faceted phenomenon. Learning processes include the acquisition of new declarative knowledge, the development of motor and cognitive skills through instruction or practice, the organization of new knowledge into general, effective representations, and the discovery of new facts and theories through observation and experimentation. Since the inception of the computer era, researchers have been, and remains, a most challenging and fascinating long-range goal in the artificial intelligence (AI). The study and computer modeling of learning processes in their multiple manifestations constitutes the subject matter of machine learning...*”

O aprendizado de máquina é organizado de forma a desenvolver a análise de sistemas de aprendizado para aumentar seu desempenho em tarefas pré-determinadas [204], a pesquisa e simulação computacional dos processos de aprendizado humano [93], a exploração teórica do espaço de possíveis métodos de aprendizado [141], [171] e algoritmos independentes [101] do domínio de aplicação.

3.6 Taxinomia do Aprendizado de Máquina

Carbonell [52] apresenta uma taxinomia na pesquisa sobre aprendizado de máquina conforme resumido na Figura 3.1. O aprendizado de máquina pode ser classificado da seguinte forma:

- A - as *estratégias de aprendizado* utilizadas. Os processos são organizados de acordo com o grau de inferência² que o sistema desenvolve na informação disponível;
- B - a *representação do conhecimento* ou habilidade adquirida pelo aprendiz³;
- C - o domínio de aplicação.

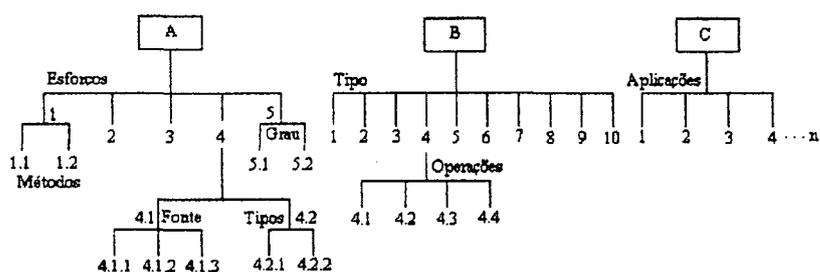


Figura 3.1: Taxinomia do Aprendizado segundo Carbonell [52]

3.6.1 Estratégias de aprendizado

A classificação baseada nas estratégias de aprendizado considera dois extremos relativos à quantidade de inferência que o aprendiz desenvolve na informação disponível: nenhuma inferência e o desenvolvimento de grande quantidade de inferência. Se um sistema computacional é programado diretamente, seu conhecimento aumenta mas não desenvolve nenhuma inferência, isto é, todos os esforços cognitivos são do programador. Reciprocamente, se um sistema, independentemente, descobre novas teorias ou inventa novos conceitos, ele desenvolverá uma substancial quantidade de inferência, derivando conceitos do conhecimento organizado de experimentos e observações.

²Kasabov [121] define a inferência como o processo de aquisição de novos fatos quando a interpretação do conhecimento existente é feita com dados correntes. O processo de inferência de novos fatos e sua manipulação é denominado raciocínio.

³O aprendiz é aqui referido como o sistema que aprende.

Assim como a quantidade de inferência que o aprendiz é capaz de desenvolver aumenta, o esforço do professor ou do ambiente externo diminui. Por exemplo, é muito mais difícil ensinar uma pessoa pela explicação de uma tarefa complexa, passo a passo, do que mostrar a maneira de fazer tarefas similares que são usualmente desenvolvidas. É mais difícil ainda programar um computador para desenvolver uma tarefa complexa do que instruir uma pessoa para desenvolvê-la pois, a programação requer a explicitação de todos os detalhes necessários, enquanto uma pessoa pode usar seu senso comum para obter certos detalhes.

Quanto a quantidade de esforços necessários pelo aprendiz e pelo professor⁴ tem-se a seguinte classificação:

1. *Implantação direta do conhecimento* ou “rote learning”: consiste de que nenhuma inferência ou outra transformação de conhecimento é necessária por parte do aprendiz. As variantes deste método de aquisição do conhecimento são:
 - 1.1 *Aprendizado por ser programado*: é construído ou modificado por uma entidade externa, não necessitando nenhum esforço por parte do aprendiz (por exemplo, estilo usual de programação de computadores);
 - 1.2 *Aprendizado por memorização*: é a memorização de datas, fatos e dados, sem nenhuma inferência obtida da informação (por exemplo, desenvolvida nos sistemas de base de dados primitivos). O termo “rote learning” (decorar) é usado primariamente neste contexto.
2. *Aprendizado de instrução* (ou, aprendizado por ser contado, narrado): é a aquisição do conhecimento é feita através de um professor ou de outra fonte organizada, tal como livro-texto, requerendo que o aprendiz transforme o conhecimento de uma linguagem de entrada a uma representação interna utilizável, e esta nova transformação é integrada com o conhecimento anterior para um uso efetivo. Portanto, o aprendiz desenvolve alguma inferência, mas uma grande fração do esforço permanece com o professor que é responsável por apresentar e organizar o conhecimento de maneira incremental, de modo que o aprendiz aumente seu conhecimento existente. Assim, uma das tarefas do aprendizado de máquina é construir um sistema que possa aceitar uma instrução ou supervisão e armazenar e aplicar este conhecimento efetivamente.
3. *Aprendizado por analogia*: é a aquisição de novos fatos ou habilidades pela transformação e aumento do conhecimento existente que possui forte semelhança com o novo conhecimento desejado. Um exemplo claro no aprendizado

⁴O “professor” é aqui referido como sendo a supervisão do aprendizado, ou agente externo ao sistema. Mais detalhes sobre o aprendizado supervisionado serão vistos na subseção 3.6.4.

humano, seria o caso de um motorista que estivesse acostumado a guiar automóveis, tivesse que aprender a dirigir camionetas ou utilitários automotivos. No aprendizado de máquina, o computador deve extrair o modo de resolver o problema por analogia a problemas semelhantes. Por exemplo, um sistema de aprendizado por analogia pode ser aplicado para converter um programa de computador existente em outro que desenvolve uma função relacionada com o programa originalmente concebido. Um fato ou parâmetro relevante deve ser recuperado da memória, este conhecimento deve então ser transformado, aplicado a uma nova situação e armazenado para uso futuro. O aprendizado por analogia requer mais inferência por parte do aprendiz do que o aprendizado por instrução e por decorar.

4. *Aprendizado por exemplos*: é dado um conjunto de exemplos e “contra exemplos” (ou exemplos negativos) de um conceito, o aprendiz induz descrição de um conceito geral que descreve todos os exemplos positivos e nenhum dos “contra exemplos”. A quantidade de inferência desenvolvida pelo aprendiz é muito maior do que o aprendizado por instrução (não existem conceitos gerais oferecidos pelo professor) e, em algumas vezes, muito maior do que no aprendizado por analogia (quando não existem conceitos similares com os quais os novos conceitos possam ser construídos). O aprendizado por exemplos foi dividido por Carbonell [52] em categorias de acordo com a *fonte* e *tipo* dos exemplos:

- 4.1 - Quanto à fonte dos exemplos:

- 4.1.1 - A fonte é um *professor* que conhece o conceito e gera seqüências de exemplos mais significativos quanto for possível. Se o professor também conhece (ou infere) o estado do conhecimento do aprendiz, os exemplos podem ser selecionados para otimizar a convergência do conceito desejado.

- 4.1.2 - A fonte é o *próprio aprendiz*. O aprendiz conhece seu próprio estado de conhecimento mas não sabe exatamente o conceito a ser adquirido. Portanto, o aprendiz pode gerar exemplos (existindo uma entidade externa tal como, ambiente ou professor, que classificará estes exemplos como positivos ou negativos) baseados na informação que ele acredita ser necessária para discriminá-los das descrições contendo tais conceitos.

- 4.1.3 - A fonte é o *ambiente externo*. Neste caso, a geração de exemplos é operacionalmente um processo aleatório e o aprendiz conta com observações relativamente não controladas.

- 4.2 - Quanto ao tipo de exemplos:

- 4.2.1 - Disponibilidade somente de exemplos positivos; exemplos positivos oferecem modelos para a aquisição de novos conceitos mas não previnem uma generalização⁵ excessiva dos conceitos adquiridos. Esta generalização excessiva pode ser evitada considerando um conjunto mínimo de generalizações necessárias ou pela restrição, *a priori*, no domínio do conhecimento, o conceito a ser inferido [52].
- 4.2.2 - Disponibilidade de exemplos positivos e negativos. Enquanto os exemplos positivos forçam uma generalização, os exemplos negativos previnem a generalização excessiva.
5. *Aprendizado por observação e descoberta*⁶: é uma forma de aprendizado que não possui um professor externo, isto requer um esforço de desenvolvimento de inferência muito maior pelo aprendiz do que qualquer outra forma de aprendizado. Não existem conjuntos de exemplos de um conceito particular, os exemplos não podem ser classificados como positivos ou negativos. Assim uma ênfase é dada quanto ao grau de interação com o ambiente externo:
- 5.1 - *Observação Passiva*: é o caso no qual o aprendiz classifica as observações de múltiplos aspectos do ambiente;
- 5.2 - *Experimentação Ativa*: consiste na perturbação do ambiente (feita pelo aprendiz) e na observação dos resultados de suas perturbações. A experimentação pode ser aleatória, dinamicamente focalizada de acordo com um critério geral de interesse ou guiado por restrições teóricas. O sistema adquire conhecimento e faz hipóteses teóricas para confirmar ou não estas teorias, portanto explora seu ambiente aplicando diferentes estratégias de observação e experimentação que necessita. Esta forma de aprendizado envolve a geração de exemplos para testar suposições ou conceitos parcialmente adquiridos.

Num sistema por descoberta, para que se associe aprendizado ao mesmo, necessita-se de memorização e um esquema de indexação. No computador

⁵A palavra generalização refere-se geralmente a extensão de um princípio ou de um conceito a todos os casos a que se pode aplicar. É o processo pelo qual se reconhecem características comuns a vários objetos singulares, daí resultando quer a formação de um novo conceito ou idéia, quer o aumento da extensão de um conceito já determinado que passa a cobrir uma nova classe de exemplos.

⁶Também denominado como aprendizado não supervisionado conforme é mostrado em outra classificação na subseção 3.6.4.

(por exemplo, na descoberta de um teorema) não é necessário um programa de aprendizado em separado, pois a prova do teorema pode ser obtida e transportada para o papel. Entretanto, esta prova pode ser armazenada pelo computador para ser usada num trabalho subsequente. Um dos primeiros programas de aprendizado para computadores foi o “Logic Theorist” de Newel e Simon [157]. Um dos problemas encontrados neste programa é que este memorizava somente o teorema e não sua prova pois seu ponto de partida era somente os axiomas. Segundo Simon [200] se também fosse dado ao programa a capacidade de “aprender decorando” (aprendizado somente por memorização), ele poderia usar novos teoremas ao longo dos axiomas encontrados e memorizar a prova.

Um sistema de aprendizado pode adquirir regras de comportamento, descrição de objetos físicos, heurísticas na resolução de problemas, taxinomias de classificação em espaço amostrais e muitos outros tipos de conhecimento úteis no desenvolvimento de uma ampla faixa de tarefas.

3.6.2 Representação do conhecimento

A classificação quanto à representação do conhecimento, ou quanto ao tipo de conhecimento adquirido foi dividida por Carbonell [52] como:

1. *Ajuste de Parâmetros em Expressões Algébricas*: O aprendizado neste contexto consiste no ajuste numérico de parâmetros ou coeficientes em expressões algébricas de uma forma funcional fixa de modo a obter um desempenho desejado. Um exemplo é encontrado no Perceptron [152], [153],[188] onde é feito ajuste dos coeficientes dos pesos para reconhecimento de padrões bidimensionais ⁷
2. *Árvores de decisão*: Alguns sistemas adquirem árvores de decisão para discriminar classes de objetos. Os nós de uma árvore de decisão correspondem a atributos dos objetos selecionados e as extremidades correspondem a valores alternativos pré-determinados para estes atributos. As folhas das árvores correspondem aos conjuntos de objetos com uma classificação idêntica. Exemplos são encontrados em sistemas especialistas para auxílio à decisão.
3. *Gramáticas Formais*: Uma gramática formal é uma gramática que define quais são as construções válidas nessa gramática através de um método semelhante

⁷Uma crítica a este modelo é encontrada em [152] [153], o que causou uma histórica polêmica nas pesquisas em IA. Mais detalhes podem ser encontrados em [36].

ao método dos sistemas formais. Nos sistemas formais têm-se um vocabulário e um conjunto de regras de derivação. No caso de uma gramática formal têm-se vocabulários (terminal, não terminal), uma palavra de início e regras de derivação (ou regras de produção). As gramáticas formais são denominadas gramáticas gerativas pois através de uma gramática pode ser gerado as construções (palavras ou frases) de uma linguagem formal. Pode-se dizer que para reconhecer se uma palavra pertence ou não a uma gramática formal existem vários métodos, dentre os quais, através da utilização da própria definição de gramática, é verificado se a palavra é gerada ou não. Este método, no entanto, não tem uma maneira direta para se saber como uma palavra pode ser deduzida. Um método mais direto é classificar esta linguagem através de uma certa hierarquia (tipo 0, 1, 2 e 3), conforme proposto por Chomsky [113], e associar a cada um destes tipos de linguagens, que são linguagens que possuem certas restrições, um tipo de máquina reconhecedora. E a cada tipo de linguagem corresponde uma máquina reconhecedora. Um tipo de linguagem são as expressões regulares, reconhecidas pelos autômatos finitos. As gramáticas livres de contexto são reconhecidas pelos autômatos de pilha. As gramáticas sensíveis ao contexto necessitam de um autômato linear limitado para acesso a todo conteúdo da pilha. E uma linguagem na qual não existem restrições com relação às regras de produção, o reconhecimento é feito através de uma máquina equivalente à máquina de Turing (linear e ilimitada).

4. *Regras de Produção:* As regras de produção são amplamente utilizadas na representação do conhecimento pela simplicidade e interpretação. Uma regra de produção é o par condição-ação $\{C \Rightarrow A\}$. Se todas as condições da regra são satisfeitas, então a seqüência de ações é executada. As quatro operações básicas nas quais as regras são adquiridas e refinadas são:
 - 4.1 - *Criação:* Uma nova regra é construída pelo sistema ou adquirida de uma entidade externa;
 - 4.2 - *Generalização:* Diminuição das condições, que tornam-se menos restritivas facilitando a aplicação em grande número de situações;
 - 4.3 - *Especialização:* Algumas condições são adicionadas ao conjunto de condições ou são feitas restrições nas condições existentes para aplicação em situações específicas;
 - 4.4 - *Composição de regras:* Duas ou mais regras podem ser aplicadas em seqüência para compor outra regra, facilitando a eliminação de redundâncias de condições e ações.

5. *Expressões da Lógica Formal e Formalismos relacionados*: As expressões da lógica formal são utilizadas para formular descrições de objetos individuais (entrada para o sistema de aprendizado) e para formular os conceitos resultantes (saída do sistema). Os componentes são preposições, predicados, variáveis valoradas, etc.
6. *Grafos e Redes Semânticas*: Em muitos domínios, os grafos (conceituais e semânticos) oferecem uma representação mais conveniente do que as expressões lógicas. Algumas técnicas de aprendizado exploram a transformação de grafos e esquemas para comparar e indexar o conhecimento eficientemente.
7. *Molduras e Roteiros*: Oferecem unidades de representação do conhecimento mais simples do que expressões lógicas e regras de produção. As molduras são coleções de entidades rotuladas - campos ("slots") - cada qual contendo uma certa descrição na representação. Por exemplo, no aprendizado por experiência, o sucesso passado, as alternativas não testadas, as causas de falhas entre outras informações podem ser registradas e comparadas na indução e refinamento de várias regras de comportamento. Os esquemas oferecem o formalismo adequado. Um roteiro é uma estrutura de dados que descreve, de modo estereotipado, uma seqüência de acontecimentos num contexto particular. Assim, como numa moldura o roteiro contém campos, aos quais podem ser associados valores por ignorância ("default") [36].
8. *Codificação de procedimentos em programas*: O objetivo de muitos sistemas de aprendizado é adquirir a habilidade de transportar um processo específico eficientemente, ao contrário de raciocinar sobre uma estrutura interna deste processo. Uma codificação de procedimentos pode incluir habilidades motoras, seqüência de instruções e outras características humanas "compiladas" por exemplo, para manipulação de robôs.
9. *Taxinomias*: O aprendizado por observação pode resultar da estruturação global do domínio dos objetos em uma hierarquia ou taxinomia. O aglomerado⁸ ("clustering") das descrições em novas categorias e a formação de uma classificação hierárquica necessitam de um critério de relevância dos objetos para sua classificação.
10. *Representações Múltiplas*: Alguns sistemas de aquisição do conhecimento utilizam diversos esquemas de representações para o novo conhecimento adquirido.

⁸A tradução aqui utilizada foi contribuição pessoal do Prof. Dr. Luiz Muniz Barreto - Consultor do Dicionário Aurélio, RJ, Novembro, 2001.

É necessário, assim, uma combinação apropriada, aplicáveis às diferentes formas do conhecimento a ser adquirido.

3.6.3 Domínio de aplicação

A classificação do domínio de aplicação envolve diversas áreas tais como Agricultura, Química, Jogos, Reconhecimento de Imagens e Voz, Matemática, Diagnóstico Médico, Música, Física, Processamento de Linguagens Naturais, Planejamento e Resolução de Problemas, entre tantas outras. Entretanto, são discriminadas aqui as de interesse neste estudo:

1. *Modelagem Cognitiva*: O aprendizado de máquina é amplamente aplicado na Modelagem dos Processos Cognitivos com a sobreposição das diversas áreas: Filosofia Computacional, Ciências Cognitivas, IA, etc.
2. *Programação de Computadores*: O aprendizado de máquina é utilizado para aumentar a eficiência no desenvolvimento e na melhoria da qualidade de programas;
3. *Educação*: Na Educação, o desenvolvimento de sistemas de Ensino Inteligente tem encontrado aplicações de modo a aumentar a eficiência do processo Ensino-aprendizagem auxiliado por Computador;
4. *Sistemas Especialistas*: Especificamente, os sistemas de especialistas dedicados ao Ensino é área de interesse desta pesquisa;
5. *Métodos Gerais*: Domínio não específico ...

Ampliando a classificação de Carbonell [52], pode-se classificar o aprendizado sob as mais diversas formas. Nas próximas seções serão dadas classificações quanto à retroação do mundo (encontrada em [36]), quanto à finalidade do aprendizado, quanto à interação com o meio.

3.6.4 Classificação segundo a retroação do mundo

Um modo de classificação do aprendizado adotado por Barreto [36] é pela presença ou ausência de retroação explícita do mundo exterior. Uma retroação explícita significa que em certos intervalos de tempo um professor assinala erros e acertos. A ausência do professor é o caso no qual a retroação não é explícita.

Aprendizado Supervisionado:

No Aprendizado Supervisionado, o sistema é treinado com o auxílio de um professor ou um treinador que indica explicitamente um comportamento bom ou ruim [36]. Um exemplo é o aprendizado supervisionado das Redes Neurais. Para tanto, o sistema deverá possuir pares de entrada e saída, ou seja, um conjunto de entradas e um conjunto com as saídas desejadas para cada entrada. Toda vez que for apresentada à rede uma entrada, deverá ser verificada a saída obtida (gerada a partir de cálculos efetuados com os pesos que a rede possui) confere com a saída desejada para àquela entrada. Sendo diferente, a rede deverá ajustar os pesos para que armazene o conhecimento desejado. Essa iteratividade do treino deverá ser repetida com todo o conjunto de treinamento (entradas e saídas), até que a taxa de acerto esteja dentro de uma faixa considerada satisfatória.

Aprendizado Não Supervisionado:

Este tipo de aprendizado também é conhecido como aprendizado de auto-organização. Esse aprendizado não requer saídas desejadas e, por isso, é conhecido pelo fato de não precisar usar “professores” para seu treinamento. No caso de uma rede neural, são usados para treinamento apenas os valores de entrada. A rede trabalha essas entradas e organiza-se de modo que as classifique, usando, para isso, seus próprios critérios. Esse tipo de rede utiliza os neurônios como classificadores, e os dados de entrada, como elementos para classificação. O processo de classificação fica por conta da rede neural e o seu algoritmo de aprendizado. A auto-organização demonstrada nas Redes Neurais não-supervisionadas envolve o processo de competição e o processo de cooperação entre neurônios da rede. Muitos pesquisadores têm utilizado esse tipo de rede como detector de características, dada a sua capacidade de aprender a discriminar estímulos ocorrendo em partes especialmente diferentes [36].

3.6.5 Classificação quanto a finalidade do aprendizado

- *Auto-associador*: Uma coleção de exemplos é apresentada ao sistema, o qual supõe-se memorizar os exemplos. Depois, quando um destes exemplos for novamente apresentado de modo deteriorado, supõe-se que o sistema restitua o original sem deterioração.
- *Hetero-associador*: É uma variante do auto-associador que memoriza um conjunto de pares. O sistema aprende a reproduzir o segundo elemento do par mesmo que o primeiro seja apresentado contendo pequenas alterações. Este hetero-associador é também conhecido como reconhecedor de padrões, onde o

primeiro elemento apresentado é o elemento a ser reconhecido e o segundo é um elemento do conjunto de padrões considerado [32].

3.6.6 Classificação quanto a interação com o meio

Nota-se, assim, que as classificações de aprendizado se sobrepõem. Uma classificação interessante pode ser feita de acordo com a interação do indivíduo com o seu meio.

- *Aprendizado com retroação do meio*: O aprendiz interage com o meio e recebe informações do meio.
- *Aprendizado sem retroação do meio*: Neste caso, o aprendiz só recebe informação do meio.

Esta classificação é particularmente útil nos processos de Ensino-aprendizagem nos quais existem além de contextos⁹, as interações dos indivíduos com o meio exterior (o professor). É uma forma de aprendizado supervisionado (retroação do meio). Esta classificação é interessante porque nela pode-se analisar as projeções.

3.6.7 Classificação quanto ao tipo de conhecimento adquirido

O conhecimento pode ser dividido em três tipos: *conhecimento fatural*, *conhecimento dedutivo* e *conhecimento hábil* (ou habilidade) [34].

O conhecimento fatural compreende os conhecimentos atômicos sobre um determinado assunto. Por exemplo, em geografia descritiva, o conhecimento fatural inclui o conjunto de dados sobre países, sobre relevo, etc. No estudo de línguas, o conhecimento do vocabulário pode ser considerado fatural.

O conhecimento dedutivo é o conhecimento que se deve dispor em um determinado assunto para manipular conhecimentos fatuais. Assim, é possível, usando conhecimentos dedutivos, a partir de um conjunto de conhecimentos fatuais, deduzir outros conhecimentos fatuais. Pode-se dizer, ainda, que o conhecimento dedutivo permite, usando conhecimentos fatuais explícitos, explicitar conhecimentos que estejam sobre a forma implícita. No caso da densidade de população seria: “para saber a densidade de população de um país divide sua população por sua superfície”. No caso de uma língua estrangeira (inglês) uma tal regra seria: “para responder negativamente a uma frase que comece por “do you” basta dizer “No, I don’t”. Ainda no caso de línguas, o conhecimento de regras de gramática permite a partir do conhecimento explícito do vocabulário formar uma frase, pode ser considerado conhecimento dedutivo.

⁹Este assunto é abordado no Capítulo 4.

Habilidade é o “meta-conhecimento” (conhecimento de como usar o conhecimento) sobre o uso do conhecimento dedutivo. No caso de uma língua seria o conhecimento hábil de escrever bem. Uma boa leitura sobre Conhecimento pode ser encontrada em Barreto [34], [36].

Em relação ao Conhecimento, encontra-se ainda conceitos de *Conhecimento Superficial* e *Conhecimento Profundo*, ambos oriundos de trabalhos que utilizam conceitos da Física qualitativa [60], [89], [128], [129], [185], [191]. O conhecimento superficial é o conhecimento é fatorial, sendo ignorada a teoria [123], [124], que o suporta. Assim, dizer que “um avião pode voar” encerra somente conhecimento superficial, pois os mecanismos que possibilitam o voo não estão presentes. Costuma-se representar o conhecimento superficial por meio de fatos e regras de manipulação destes fatos [49]. Em contrapartida, o conhecimento profundo frequentemente, em muitos domínios, é explicitado com o que se denomina “leis”. Essas leis são modelos do mundo real que são válidos dentro de um certo número de condições. Não são mais que modelos, mas considera-se que são tão bem conhecidos que se diz que são de conhecimento profundo. O conhecimento profundo apoia-se numa teoria bem estabelecida a qual permite um raciocínio causal [31], [115], [116], [118]. Por exemplo, a lei de Newton, “força é igual ao produto da massa pela aceleração”, contém um conhecimento profundo [40], [136].

3.7 Aprendizado: faceta importante da Inteligência

O aprendizado é fundamental para a sobrevivência de qualquer espécie e tem características que podem ser medidas quanto à sua extensão (sua generalidade e sua profundidade). Estudos recentes [36], [86], [87], [112] têm procurado mostrar que a complexidade do processo de aprendizado não é acidental mas é um produto adaptativo da evolução. A Inteligência Evolutiva individual é relativa ao indivíduo e a inteligência coletiva está relacionada com a espécie. No aprendizado de máquina a inteligência coletiva é a Inteligência Artificial Distribuída.

3.7.1 O Especialista e o Generalista

Um ensinamento profundo é aquele que se proclama, hoje em dia, dizendo ser preciso formar grandes especialistas, no qual o indivíduo escolhe os assuntos para sua mente e procura aprender “tudo” sobre aquele assunto. Realmente, tendo em vista a grande quantidade de coisas descobertas e o tempo limitado da vida humana, é uma forma de se obter uma grande quantidade de conhecimento sobre um determinado assunto e juntá-los a outros já conhecidos, que é caso do Aprendizado do

Especialista. Outro tipo de aprendizado é o do *Generalista* (ou adaptável), que é capaz de aprender a sobreviver em ambientes diversos. Essas são as duas correntes básicas do *Aprendizado Evolutivo*. Uma pergunta se faz, qual dos dois (especialista ou generalista) é mais inteligente?

A esta pergunta não se tem resposta, pois são duas inteligências diferentes que não podem ser medidas por um fator único. A Evolução do aprendizado vai de acordo com um ambiente extremamente fechado e invasável (favorendo o especialista) ou mutável, que favorece o Generalista. Quando um ser aprende e ganha na competição com seus semelhantes pela especialização do seu conhecimento, se o ambiente muda, a tendência é a sua morte. Na natureza encontram-se vários exemplos de seres especialistas (como o urso Panda) e generalistas (baratas, ratos, etc.). Um exemplo de aprendizado evolutivo é o aprendizado da barata que é capaz de sobreviver nos ambientes mais inóspitos, desenvolvendo até mesmo uma circulação sanguínea (o denominado “sangue de barata”) adaptada. Na verdade, a barata não tem alta performance em nada mas a evolução lhe ensinou a resolver problemas de sobrevivência em ambientes adversos. O generalista, em face das mudanças ambientais consegue modificar seus parâmetros.

Considerando-se o cérebro humano como uma coleção muito grande de programas acumulados ao longo da vida, pode-se dizer que estes “programas cerebrais” são modificados e repetidos pelo processo de aprendizado evolutivo, que não necessita de representações internas do conhecimento. Supondo-se que fosse possível abrir e programar o cérebro diretamente, de forma a escrever, depurar e modificar os programas cerebrais, seria possível, então, aprender uma série de coisas sobre o código interno, as representações internas do conhecimento e as habilidades desenvolvidas. Talvez uma pessoa saiba como o seu conhecimento está organizado em seu cérebro mas não sabe como este conhecimento está organizado no cérebro de uma outra pessoa. Também não é possível um “compartilhamento dos programas cerebrais” o que poderia diminuir a lentidão do aprendizado animal.

Outra característica importante do aprendizado animal é que nenhum professor ou aprendiz tem uma representação interna detalhada dos dados ou procedimentos. Um exemplo de aprendizado evolutivo com aplicação em Redes Neurais (modificação da topologia e dos valores das conexões sinápticas de modo a fazer uma rede tornar-se apta a resolver um problema) pode ser encontrado em [36].

3.7.2 O problema da medida métrica da inteligência

A inteligência não pode ser medida com testes do tipo “Quociente de Inteligência” (QI). O QI é medido através de uma técnica de comparação de dados espaciais de po-

pulações sobre as mesmas condições (por exemplo, crianças calmas da mesma idade) não podendo ser aplicado para todos porque não considera condições emocionais¹⁰ (emoções, estresse, diferenças de idade). Mesmo a idéia do QI emocional (determinadas emoções influenciam o raciocínio dedutivo) não considera um conjunto de outras coisas que influenciam o desempenho do comportamento animal.

O fato é que a inteligência não pode ser medida por um número porque não é uma entidade no espaço métrico. O espaço métrico é aquele que pode ser medido, ou seja, existe o conceito de distância. Um exemplo de espaço que não é métrico pode ser encontrado na Física (Mecânica dos Gases), onde têm-se os eixos de Pressão, Volume, Temperatura no espaço P,V,T e não tem sentido falar de distância entre dois pontos.

As coisas mais elementares da vida não são métricas, o ser humano é que têm uma tendência a “metrizar” tudo, reduzindo as coisas a números. A inteligência não pode ser reduzida a um número por mais tentativas que se faça. Alguns pesquisadores falam até em medidas de inteligência de máquinas. Também no aprendizado de máquina pode-se ter conceitos, idéias, aproximações e comparações, mas sempre que estiver falando de inteligência, deve-se dizer sobre qual é o critério, qual é o contexto.

Tal fato leva também ao conceito de “emergência da inteligência”. Sabe-se que para se possuir um comportamento emergente, este deve ter um observador (ou seja, deve ser observável) de acordo com uma escala diferente daquela da existência do fenômeno. Por exemplo, a cor marrom de determinado objeto visto por alguém é o comportamento emergente dos elétrons que compõem o objeto. A queda de vários níveis enérgicos orbitais dos elétrons é que produz a cor marrom vista pelo observador, ou seja um comportamento emergente da cor marrom. Se não existe observador, não existe comportamento emergente e este por sua vez ocorre em relação aquele observador. O que emerge para um observador difere do que emerge para outro, sendo necessário que o observador seja fixo. Um exemplo de emergência no aprendizado de máquina é o caso das Redes Neurais. Elas podem ter o comportamento emergência da inteligência para responder como um sistema especialista, como uma retina artificial, etc. No caso dos seres humanos, pode-se ter, por exemplo, uma pessoa explicando algo (matemática, um fato, etc.) a outra pessoa (que é o observador). Isso é o comportamento emergente dos neurônios da primeira pessoa.

O comportamento emergente é, então, algo que produz uma sensação cognitiva num observador, não existindo comportamento emergente sem haver cognição, o que conduz a uma definição neurológica do que é cognição: é o comportamento emergente de um conjunto de neurônios.

¹⁰Inteligência Emocional.

3.7.3 A Inteligência Humana é uma Função Computável?

O aprendizado é a base da inteligência. Uma pergunta se faz: a inteligência humana é uma função computável ?

Com a finalidade de mostrar a potencialidade dos métodos formais tenta-se aqui responder a essa pergunta formalmente.

Definição 3.7.1 *Uma função é dita computável (ou decidível) se é possível calcular seu valor para todos os elementos de seu domínio de definição. No caso contrário, ela é dita não computável.*

Assim, supondo uma relação binária, uma função é computável (ou decidível) se para qualquer valor de seus argumentos pode-se saber se o valor desta função é verdadeiro ou falso. Ela será não computável (no caso, indecidível) se existirem valores no seu domínio para os quais não se pode saber, a priori, se ela é ou não computável¹¹. Assim, para saber se a inteligência humana é uma função computável tem-se que verificar dois pontos:

1. Se a inteligência humana pode ser modelada por uma função;
2. Se essa função pode ser provada como sendo computável.

O primeiro ponto, para considerar-se a inteligência humana como uma função define-se a inteligência humana:

Definição 3.7.2 *A inteligência é a capacidade de resolver problemas.*

Se for aceita esta definição para inteligência, determinando o que se chama de função inteligência, F_i , como a “capacidade de resolver problemas”, tem-se que o domínio desta função é o conjunto de problemas e seu contradomínio é a solução destes problemas.

Definição 3.7.3 *Um problema é a terna obtida do conjunto de dados, conjunto de soluções e uma condição q que caracteriza a solução.*

$$P = \{D, S, q\}$$

Isto significa que a F_i a ser definida, tem, por domínio, um conjunto de ternas e, por contra-domínio, a solução do problema, ou seja, a condição q deste conjunto de ternas. O problema da computabilidade de F_i reduz-se, então, em saber se: dado

¹¹Mais detalhes sobre computabilidade de funções podem ser obtidos em Barreto [38], [36].

um conjunto de problemas é sempre possível identificar a condição através desta F_i . Então, desta definição, diz-se que a inteligência é uma função F_i com a condição q a descobrir.

Em suma:

- ◇ F_i , função inteligência definida como a capacidade de resolver problemas;
- ◇ D conjunto de dados do problema;
- ◇ S conjunto de soluções do problema;
- ◇ q uma relação binária, $q \subset D \times S$, condição que caracteriza a resolução do problema.

É necessário saber se é sempre possível definir F_i . Neste caso, deve-se verificar como é apresentado o problema pois o conjunto de problemas (domínio) pode ser apresentado de vários modos, por exemplo, por enumeração exaustiva, declarativamente, por procedimentos e por exemplos.

Teorema 3.7.1 *Se o domínio da F_i for definido em cima de um conjunto finito, a função é computável.*

A inteligência humana pode ser uma função computável quando o conjunto de tarefas para resolução do problema for finito, ou seja, o domínio for definido em um conjunto finito, pois será possível experimentar todas as tarefas e verificar se cada uma delas resolve ou não o problema. Em outras palavras, supondo que o número de coisas a aprender é finito, pode-se então, aprender todas as coisas. No caso de um conjunto finito, tem-se a enumeração finita dos elementos e, achar o elemento no contra-domínio, é apenas examinar neste conjunto finito qual é o elemento condição.

É possível computar o valor da função (binário) com respostas de sim (verdade) ou não (falso) à pergunta: resolve ou não a tarefa? Dizer que a função é computável não significa que ela seja capaz de resolver todas as tarefas finitas e se ela resolve ou não resolve.

Teorema 3.7.2 *Se o domínio da F_i for definido, determinado através de exemplos, a função não poderá ser determinada pois existirá mais de função satisfazendo estes exemplos e, portanto a função é não computável.*

Neste caso, o problema não é completamente definido para todos os valores de seus dados. Se o conjunto de problemas for infinito (definido por exemplos ou por procedimentos declarativos), a inteligência humana só poderá ser aproximada com

o grau de precisão desejada de acordo com o número de exemplos que se dispõe. Então, a solução não é única: todas as funções que sejam iguais dentro da região em que o problema é definido são válidas.

Teorema 3.7.3 *Quando a F_i for definida por procedimentos. Existirão procedimentos que levarão à solução do problema e os que não irão garantir a solução do problema. Assim, a F_i é não computável.*

O problema da computabilidade [38], [65] da F_i se reduz ao Problema da Parada. Sempre haverá dados para os quais não se sabe se conduzirão a uma solução algorítmica do problema. Assim, quando o problema for definido através de procedimentos (que gerem condições a partir de dados e resultados), existirá o Problema da Parada. Se o problema for definido de maneira declarativa deverá existir um transformador declarativo para procedimentos de operações, que por definição é não computável. No caso da F_i se o problema for definido por procedimento não existe critério para saber se ele pára ou não pára, ou seja, não sabe se ele é computável. Diante disso, tem-se a resposta para pergunta:

A inteligência humana não é uma função computável!

Dentro do computador só existem sistemas formais e pode-se dizer que não existe “realmente” uma Cognição. Assim, o aprendizado de máquina está intrinsecamente relacionado com a manipulação e representação simbólica. O resultado do que foi aqui obtido de maneira formal vem coincidir com trabalhos de pessoas como Penrose [174], que não acreditam na IA pura, ou então pessoas como Dreifuss [70], [71] que acham que é impossível fazer a máquina ser inteligente. Aqui apresentou-se uma demonstração formal das conjecturas desses autores. Existem outros autores que acreditam ser possível fazer uma cópia, de maneira perfeita, da inteligência natural [51], [52], [200]. Justificar essa possibilidade seria examinar quais são os pontos de partida da demonstração dos teoremas que deveriam ser alterados. Um ponto merecedor de alteração é que todo raciocínio foi feito de modo algorítmico. Pode-se dizer, então, que se a inteligência fosse implementada em uma máquina de modo não algorítmico talvez fosse possível implementar esta “inteligência artificial” na máquina porque a validade da demonstração está fora validade da demonstração de modo não algorítmico. Na demonstração, em modo algorítmico, existe o conceito de etapa (senão não tem sentido falar que é computável ou não computável). No modo não algorítmico, como no caso das Redes Neurais contínuas, não existe o conceito de etapa. Conseqüentemente, no domínio do tempo contínuo, poderia haver o “sistema inteligente”, o que isso coincide com o cérebro biológico porque a informação é transmitida com valores de frequência e processamento paralelo. Nas Redes Neurais

contínuas implementadas em pastilhas analógicas [23], [53], [145], [146], [147], a IA pura estaria muito mais próxima pois, estes modelos, são plausíveis biologicamente.

Isto leva ao fato de que não se pode querer imitar a inteligência animal. O que pode ser feito, no máximo, são aproximações de coisas inteligentes. Assim, é preciso contar com a própria inteligência humana para que a mesma cresça sozinha, ou seja realimentando-se, por limitações que existem na própria natureza...

Capítulo 4

Sistemas Formais e Teoria das Categorias

“... any idea or problem or body of knowledge can be presented in a form simple enough so that any particular learner can understand it in a recognizable form.”

Bruner

Neste Capítulo serão apresentados os conceitos introdutórios dos Sistemas Formais na Computação, necessários à compreensão do trabalho. Será apresentada a Teoria das Categorias [26], [130] apenas como uma recordação de conceitos elementares pois o objetivo aqui não é desenvolver os conceitos de Teoria das Categorias mas apresentar aqueles que serão efetivamente utilizados. Inicialmente, de modo a familiarizar o leitor nesta abordagem, será mostrada a Categoria dos Conjuntos e, em seguida serão citados alguns exemplos. Em particular, de interesse neste trabalho será caracterizada a Categoria dos Conceitos e a Categoria dos Autômatos.

4.1 O que são sistemas formais?

O Dicionário Aurélio [79] apresenta seis significados para o termo *Formal*:

1. Relativo à forma;
2. Evidente, claro, manifesto, patente;
3. Preciso, próprio, genuíno;

4. Que não é espontâneo; que se atém às fórmulas estabelecidas; convencional;
5. Que é amigo de formalidades, de etiquetas; formalista.
6. Filos. Relativo às leis, às regras ou à linguagem próprias de determinado domínio do conhecimento, e que se consideram independentemente do conteúdo, da matéria ou da situação concreta a que se aplicam.

Embora todos os significados se apliquem a esta tese em sua essência, ressalta-se que:

- Formal se refere à forma e, portanto, sistemas formais, são sistemas de manipulação de formas, sem preocupação do que significam no mundo real;
- A essência de um sistema formal é, portanto, sua sintaxe. Existe a semântica formal, mas seu estudo foge ao escopo deste trabalho.

4.2 Origens dos sistemas formais

A primeira notícia que se tem de um sistema formal são os trabalhos de Euclides (300A.C.) [75], [76]. Estes trabalhos organizam e sistematizam todo o conhecimento da época com relação à Geometria e são conhecidos sob o nome de **Elementos**. Pela primeira vez a apresentação é feita através de axiomas, definições, postulados, teoremas e demonstrações. Neste trabalho encontra-se as raízes dos conceitos de termos primitivos e de outros mencionados atualmente.

O arcabouço básico dos sistemas formais devem-se a René Descartes (1596-1650) e a Leibniz (1646-1716) sobre linguagens e alfabetos. Frege (1848-1925), Peano (1858-1932), Whitehead (1861-1947) e Bertran Russel (1872-1970) e, finalmente, Wittgenstein (1889-1951) criaram a formalização como se costuma apresentar nos dias de hoje [139], [138].

Na construção de um sistema formal deve-se concentrar atenção na forma com que se trabalha. As Linguagens Naturais, usadas entre seres humanos para se comunicarem, possuem ambigüidades que impedem seu uso para este propósito. Portanto, torna-se necessário dar um passo na direção de evitar estas ambigüidades, o que é feito usando uma linguagem constituída por um conjunto bem definido de símbolos e de regras de derivação, permitindo construir novos objetos a partir daqueles que se dispõe.

4.3 Métodos Formais, Engenharia de Programas e Categorias

Os métodos formais são técnicas com fundamentos matemáticos usadas para descrever e analisar propriedades de sistemas [105], [192]. Um método formal possui três características essenciais:

1. **Sistemas Formais:** consiste no uso de sistemas formais, isto é, de linguagens formais com sintaxes bem definidas, semântica e sistemas de provas. Os sistemas formais para especificação de programas podem ser funções lógicas parciais, Teoria dos Conjuntos, Teoria das Categorias e sistemas de provas.
2. **Técnica de desenvolvimento:** é a idéia de refinamento, na qual uma implementação é produzida por uma especificação através da aplicação de um número de passos de desenvolvimento, cada qual focalizando a compreensão das decisões do projeto. Isso envolve a captura dos requisitos do sistema em uma especificação abstrata usando uma linguagem de especificação.
3. **Técnica de Verificação:** de maneira a assegurar que a série de passos de refinamento preservem a correção no topo do nível de especificação, existe uma obrigação de provar que cada refinamento modela corretamente a especificação anterior.

O propósito de um método é guiar o entendimento de tarefas específica ou processos de desenvolvimento [111]. As principais características dos métodos formais englobam:

1. Os métodos formais oferecem notações para especificação funcional das decisões a serem abstraídas para caracterização dos requisitos ou da implementação. Uma linguagem de especificação é usada para este propósito [205].
2. A noção de abstração [104] é essencial na aplicação do método formal. O primeiro passo é produzir uma especificação abstrata caracterizando as propriedades essenciais do problema e declarando o que é necessário e como isso pode ser atingido.
3. O processo de refinamento é progressivo quando desenvolvido através do projeto com a implementação, na qual os detalhes específicos são introduzidos sistemática e progressivamente.
4. As obrigações de provas oferecem um suporte para as atividades de verificação e validação. O foco é dado rigorosamente ou formalmente nas questões

críticas relativas à consistência e a correção da especificação e dos passos de refinamento.

5. A decomposição conduz à quebra de grandes especificações em componentes os quais podem ser refinados, independentemente, através da composição de combinações que devem satisfazer a uma especificação maior.
6. São oferecidas linhas guias para avaliação de especificações.

Um método formal pode ser visto como o uso de sistemas formais, o uso de abstração, o refinamento e a geração de provas. Os métodos formais complementam outras abordagens [88], [110], [125], [137] de desenvolvimento de sistemas. A Teoria das Categorias atende, de forma natural, a esses quesitos. A modelagem categórica é uma ferramenta formal muito útil na Ciência da Computação, conforme encontrado no clássico trabalho de Goguen [98]:

- Formulação de definições e teorias: na Ciência da Computação é mais difícil formular conceitos e resultados do que prová-los;
- Desenvolvimento de provas: uma vez que um conceito tenha sido formulado em linguagem categórica, as provas “já acontecem” a cada passo;
- Descoberta e exploração de relações com outros campos: as formulações, suficientemente abstratas, podem revelar surpreendentes conexões com a descoberta de uma nova categoria;
- Tratamento com abstrações e representações independentes: na Ciência da Computação, pontos de vista mais abstratos são úteis porque oferecem representações independentes de detalhes complexos de representações ou implementações;
- Formulação de conjecturas e direcionamento de pesquisas: as conexões com outros campos podem sugerir novas questões no próprio campo em estudo;
- Unificação: a Computação é uma ciência muito fragmentada, com diferentes sub-disciplinas, contendo muitas direções dentro das mesmas. A Teoria das Categorias pode oferecer uma unificação conceitual na área.

A Teoria das Categorias oferece um alto nível de abstração para linguagens de descrição de arquitetura de programas. Uma importante contribuição da Teoria das Categorias para a Engenharia de Programas é ilustrar a formalização de mapeamentos entre diferentes níveis da arquitetura do programa, ou seja, é o conceito

de generalização da instanciação de componentes de baixo nível da arquitetura do programa. A existência de um funtor entre dois níveis assegura a composição, isto é, a capacidade de efetuar composições de baixo nível independente das instanciações que possam ser feitas posteriormente.

Fiadeiro [80] mostra como os conceitos elementares da Teoria das Categorias podem ser usados para formalizar a arquitetura de um programa independente do formalismo escolhido para descrição dos componentes. Isto oferece um domínio semântico para a Engenharia de Programas complementando a escolha do modelo computacional. Fiadeiro propõe uma categoria de programas, em um estilo categórico, no qual a escolha do processo de interconexão de componentes em configurações de sistemas complexos é separado da sua descrição interna.

Nelson [156] apresenta a Teoria das Categorias na modelagem formal de objetos de uma base de dados denominado modelo de produto. Os modelos de dados relacionais utilizam a teoria dos conjuntos como arcabouço formal, assegurando um rigoroso modelo matemático dos dados com suporte para manipulações. As novas gerações de modelos de base de dados são baseados no paradigma da programação orientada a objetos, cujas manipulações são mais complexas. A Teoria das Categorias oferece então um formalismo para os objetos destas base de dados. Embora qualquer teoria possa ser utilizada na modelagem de objetos de uma base de dados, os múltiplos níveis da arquitetura da Teoria das Categorias, comparada com outras teorias, torna a modelagem muito menos complexa, quando são necessários diferentes níveis para esquemas e diagramas nas bases de dados. Outra vantagem é que, como a Teoria das Categorias é baseada em diagramas, este conceito primitivo torna a modelagem mais natural com aspectos dinâmicos e estáticos. Vale ressaltar que as ferramentas categóricas de caráter diagramático (diagramas, equações algébricas, testes de consistências) são muito úteis em qualquer modelo de base de dados.

4.4 Representação de Sistemas Formais

Sistemas formais costumam ser representados por letras gregas maiúsculas. Por exemplo:

$$\Gamma, \Pi, \Sigma, \Omega, \Xi, \Psi$$

Para a representação de um sistema formal, algumas definições são feitas:

Definição 4.4.1 Símbolo: *Um símbolo é alguma coisa que pode representar qualquer outra. Um símbolo por si não tem significado.*

Exemplo 4.4.1

$$a, b, \Lambda, \Sigma, \Omega, w$$

Definição 4.4.2 *Cadeia de símbolos: Quando se tem vários símbolos sucessivamente, tem-se uma cadeia.*

Exemplo 4.4.2

$$aaabb$$

$$\Sigma\Lambda$$

Definição 4.4.3 *Um alfabeto é um conjunto finito de símbolos. Os alfabetos serão denotados por letras gregas maiúsculas. Exemplos: Φ e Σ .*

Exemplo 4.4.3 *Exemplos de alfabetos:*

- $\{a, e, i, o, u\}$ é o alfabeto constituído pelas vogais de nosso alfabeto.
- $\{I, V, X, L, C, M\}$ é o alfabeto formado pelos símbolos usados na numeração romana.
- $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ é o alfabeto formado pelos símbolos usados na numeração arábica.

Um conjunto vazio também é considerado um alfabeto. Um símbolo (ou caractere) é uma entidade abstrata básica que não é definida formalmente. Letras e dígitos são exemplos de símbolos freqüentemente usados.

Costuma-se ainda com relação a alfabetos, usar os seguintes símbolos:

Definição 4.4.4 *O conjunto de todas as cadeias finitas formadas com os elementos do alfabeto Φ é denotado por Φ^* .*

Definição 4.4.5 *Uma palavra, cadeia de caracteres ou sentença sobre um alfabeto é uma seqüência finita de símbolos (do alfabeto) justapostos.*

Exemplo 4.4.4

$$\Phi\Omegaaaa$$

Definição 4.4.6 *A cadeia ou palavra vazia, representada pelo símbolo ε , é uma palavra sem símbolo. Se Σ representa um alfabeto, então Σ^* denota o conjunto de todas as palavras possíveis sobre Σ . Analogamente, Σ^+ representa o conjunto de todas as palavras sobre Σ , excetuando-se a palavra vazia, ou seja, $\Sigma^+ = \Sigma^* - \{\varepsilon\}$.*

Definição 4.4.7 O conjunto Φ^* / ε , isto é, $(\Phi^* - \varepsilon)$ é o conjunto de todas as cadeias finitas, a partir do alfabeto, excluída a cadeia vazia, será denotado por Φ^+ .

Definição 4.4.8 Tamanho ou comprimento de uma cadeia: a função tamanho de uma cadeia é definida como a quantidade de símbolos que compõem esta cadeia, ou seja, é o número de elementos da mesma. O comprimento da cadeia μ denota-se $\rho(\mu)$ ou $|\mu|$.

Exemplo 4.4.5

$$\mu = aabb$$

$$|\mu| = 4$$

Definição 4.4.9 Prefixo, Sufixo, Subpalavra: Um Prefixo (respectivamente, Sufixo) de uma palavra é qualquer seqüência de símbolos inicial (respectivamente, final) da palavra. Uma subpalavra de uma palavra é qualquer seqüência de símbolos adjacentes da palavra.

Exemplo 4.4.6 Sendo *abcb* uma palavra sobre o alfabeto $\{a, b, c\}$.

- Se $\Sigma = \{a, b\}$, então:

$$\Sigma^+ = \{a, b, aa, ab, ba, bb, aaa, \dots\}$$

e

$$\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, \dots\}$$

- $|abcb| = 4$ e $|\varepsilon| = 0$
- ε, a, ab, abc são os prefixos da palavra *abcb* e $\varepsilon, b, cb, bcb, abcb$ são os respectivos sufixos;
- Qualquer prefixo ou sufixo de uma palavra é uma subpalavra.

Definição 4.4.10 Regra de Derivação: É uma regra de dedução (ou inferência, sem semântica) que tem um objetivo. Por exemplo, num conjunto associar um elemento a uma cadeia, é uma Regra de Derivação.

Seja o alfabeto Φ e seja $n \in \mathbb{N}$ um número natural. Uma Regra de Derivação é uma função:

$$F : \Phi^{*n} \rightarrow \Phi^*$$

Exemplo 4.4.7 *Sejam os dois elementos de $\Phi^* = \{., (xy), (car'), .\}$ Uma Regra de Derivação será:*

$$F : ((xy)(car')) \mapsto (car'(xy))$$

Que também pode ser escrita por:

$$\frac{(xy)(car')}{(car')(xy)}$$

Neste caso, não significa pegar o primeiro da lista, pois não interessa o significado. Também não é uma divisão, é dado o que está em cima e resulta o que está embaixo da barra. Se houver inversão dos elementos tem-se outra regra. Uma Regra de Derivação cria outros elementos.

Definição 4.4.11 *Um sistema formal é um par constituído por objetos e regras de derivação.*

$$\langle \Phi, D \rangle$$

Exemplo 4.4.8 *Num sistema formal, tem-se o conjunto de símbolos e o conjunto de regras de derivação. Seja o sistema formal que tem como objetos $\Phi (\{0, 1\})$ e três regras de derivação $D (\{\frac{\alpha}{0}, \frac{\alpha}{1\alpha}, \frac{\alpha}{10\alpha}\})$:*

$$\langle \Phi, D \rangle$$

é:

$$\langle \{0, 1\}, \{\frac{\alpha}{0}, \frac{\alpha}{1\alpha}, \frac{\alpha}{10\alpha}\} \rangle$$

Trata-se de um sistema capaz de gerar os números pares do sistema de numeração binário, mas esta interpretação é irrelevante para o sistema formal. Este sistema também poderia representar algo em outro domínio, como em circuitos elétricos, no qual o 1 representaria um circuito fechado e o 0 representaria o circuito aberto.

- A regra $\frac{\alpha}{0}$ poderia representar, que se não houver nada no circuito tem-se um circuito aberto;
- A regra $\frac{\alpha}{1\alpha}$ poderia representar, se tem um circuito qualquer (α) tem-se um circuito fechado em série com um circuito qualquer (1α);
- A regra $\frac{\alpha}{10\alpha}$ poderia representar, se tem um circuito qualquer tem-se um circuito aberto paralelo com um circuito qualquer.

Assim, tendo-se duas coisas com a mesma forma, uma pode representar a outra!

Definição 4.4.12 *Composição de regras de derivação. Pode-se compor regras de derivação pela aplicação sucessiva de duas regras. Assim:*

$$r_1 \cdot r_2 = r_3$$

Esta composição de regras é como uma nova regra r_3 que não aparece na definição do sistema formal.

Definição 4.4.13 *Regra identidade: uma regra que nada faz é denominada de regra identidade r_i .*

Definição 4.4.14 *A Composição de regras de derivação é associativa, pois:*

$$r_1 \cdot (r_2 \cdot r_3) = (r_1 \cdot r_2) \cdot r_3$$

Num sistema formal tem-se um **alfabeto** e **regras de derivação** que permitem chegar a todo um conjunto de elementos de uma determinada **linguagem**. Supondo-se que cada elemento da linguagem seja um teorema desta linguagem, para deduzir o teorema aplica-se regras de dedução fazendo substituições em cima do alfabeto.

4.5 Linguagens e Gramáticas Formais

Um sistema formal de grande interesse em Teoria da Computação é o das Gramáticas Gerativas [149]. Estas Gramáticas apareceram inicialmente no estudo de Linguística para uma melhor compreensão das linguagens naturais. A definição de uma linguagem por gramáticas gerativas tem grande relação com o estudo dos Sistemas Formais, podendo mesmo ser considerado como exemplo destes sistemas. O estudo das linguagens formais, sob a forma de gramáticas gerativas, é a base da teoria da interpretação e compilação. Juntamente com a Teoria de Autômatos¹, como reconhedores das palavras válidas definidas nestas linguagens, tem-se um arcabouço teórico para a compreensão e estudo de compiladores [65]. Além disso, entre as aplicações das linguagens formais estão os modelos de crescimento em biologia, modelos de evolução biológica, etc. [38].

Definição 4.5.1 *Linguagem Formal: é o conjunto de palavras sobre um alfabeto. Seja um alfabeto de referência Φ e o conjunto de objetos relativos a este alfabeto Φ^* . Uma linguagem é um subconjunto de Φ^* , isto é:*

$$L \subseteq \Phi^*$$

¹Alguns conceitos da Teoria dos Autômatos são abordados no Capítulo 5.

Quando se deseja explicitar o alfabeto escreve-se:

$$L_{\Phi} \text{ ou } L(\Phi)$$

Definição 4.5.2 *Seja um alfabeto de referência Φ e o conjunto de objetos relativos a este alfabeto Φ^* . Uma linguagem é um subconjunto de Φ^* , isto é:*

$$L \subseteq \Phi^*$$

Uma Gramática é considerada como um formalismo de geração, pois permite derivar (“gerar”) todas as palavras da linguagem que representa.

Definição 4.5.3 *Uma Gramática é uma quádrupla ordenada:*

$$\langle V, T, P, S \rangle$$

V é um conjunto finito de símbolos (alfabeto), cujos elementos são denominados símbolos *não terminais* ou variáveis;

T é um conjunto finito de símbolos *terminais* disjunto² de V ;

P é um conjunto finito de pares, denominados de regras de produção tal que a primeira componente é palavra de $(V \cup T)^+$ e a segunda componente é palavra de $(V \cup T)^*$;

S é um elemento de V denominado *variável inicial* ou sentença.

Definição 4.5.4 *Regra de Produção: Uma regra de produção (α, δ) é representada por $\alpha \rightarrow \delta$. As regras de produção definem as condições de geração das palavras da linguagem.*

Uma seqüência de regras de produção da forma $\alpha \rightarrow \delta_1, \alpha \rightarrow \delta_2, \dots, \alpha \rightarrow \delta_n$, (mesma componente no lado esquerdo), pode ser abreviada como uma única regra de produção na forma:

$$\alpha \rightarrow \delta_1 | \delta_2 | \dots | \delta_n$$

Em geral, símbolos não terminais ou variáveis de gramáticas são denotados por $A, B, C, D, E, \dots, S, T$. O símbolo inicial é denotado por S . Os símbolos terminais são denotados por $a, b, c, d, e, \dots, s, t$.

As palavras (seqüências) de símbolos terminais são denotadas por caracteres em minúsculo u, v, w, x, y, z . As palavras que possuem símbolos terminais e não terminais, são, geralmente, representadas pelas letras gregas minúsculas $\alpha, \delta, \gamma, \dots$ e representam as seqüências nas regras de produção:

²Quando os conjuntos não possuem elementos comuns, a interseção é o conjunto vazio.

$$\begin{aligned} \alpha &\longrightarrow \delta_1 \\ \alpha &\longrightarrow \delta_2 \\ &\vdots \\ \alpha &\longrightarrow \delta_n \end{aligned}$$

A aplicação de uma regra de produção é denominado derivação de uma palavra. A aplicação sucessiva de regras de produção permite derivar as palavras da linguagem representada na gramática.

Numa teoria tem-se um conjunto de axiomas que podem ser falsos ou verdadeiros. A dedução pode ser gerada pela aplicação de regras de derivação em uma seqüência de objetos. Formalmente, tem-se:

Seja a teoria Θ e uma seqüência de objetos $O = (o_1, o_2, o_3, \dots, o_s)$ obtidos sucessivamente pela aplicação das regras de derivação $R = (r_1, r_2, r_3, \dots, r_{s-1})$ de um sistema formal.

Tem-se que:

- R : dedução;
- O : passos da dedução;
- o_s : conclusão.

A potencialidade da sintaxe é ter uma representação para inúmeras coisas diferentes. Isto faz com que a utilidade de um sistema formal seja externa e ele mesmo. Assim, métodos formais podem ser aplicados em diversos domínios, dependendo do contexto, como será explicado, a seguir, na Teoria das Categorias.

4.6 Teoria das Categorias

O conceito de categoria nasceu da necessidade de formalizar o contexto de um discurso no qual tem-se essencialmente os objetos de que se fala e as ligações entre estes objetos. Pode existir, ainda, ligações que façam com que objetos diferentes pertençam à mesma categoria, que podem ser:

- **Reais:** são as categorias que existem no mundo real e podem ser representadas por categorias abstratas. Elas podem, ainda, serem consideradas interpretações de categorias abstratas. Por exemplo, *Categoria dos artigos sobre esportes*: Os objetos são os artigos. As ligações são pares de esportes.

- **Abstratas:** são entidades matemáticas, que podem ter várias interpretações. Por exemplo, Categoria dos Conjuntos (“Set”), Categoria dos Conjuntos Parcialmente Ordenados (“Poset”), Categoria dos Autômatos, Categoria dos Conjuntos Nebulosos, Categoria dos Espaços topológicos, etc.

4.6.1 Teoria das Categorias: definições

Segundo Hoare [109] “...*Category theory is quite the most general and abstract branch of pure mathematics. The corollary of a high degree of generality and abstraction is that the theory gives almost no assistance in solving the more specific problems within any of the subdisciplines to which it applies. It is a tool for the generalist, of little bene to the practitioner*”.

Para Asperti & Longo [27]: “... *Category theory is a mathematical jargon. Many different formalisms and structures may be proposed for what is essentially the same concept; the categorical language and approach may simplify through abstraction, display the generality of concepts, and help to formulate uniform definitions.*”

Scott [194] afirma que “... *Category theory is a pure theory of functions, not a theory of functions derived from sets.*”

Definição 4.6.1 *Categoria é o par (Ob, Mor) onde Ob são os objetos da categoria e Mor são os morfismos satisfazendo a:*

- *Morfismos se referem a pares de objetos; assim existe $Mor(Ob_1, Ob_2)$;*
- *Uma Composição de morfismos é um morfismo;*
- *A Composição de morfismos é associativa;*
- *Existe o morfismo identidade.*

4.6.2 Um Sistema Formal gera uma Categoria

A Figura 4.1 mostra um Sistema Formal gerando uma Categoria. Cada um dos elementos do alfabeto torna-se um objeto da categoria bem como todos os objetos são gerados pela utilização das regras de inferência ou dedução nos objetos da categoria. Assim, numa linguagem formal, tem-se como objeto as palavras da linguagem e os morfismos associam, dada uma palavra da linguagem, outra palavra dessa linguagem.

A Figura 4.2 mostra a representação dos objetos Ob_1 , Ob_2 , Ob_3 e morfismos Mor_{11} (identidade), Mor_{12} , Mor_{13} e Mor_{23} numa Categoria.

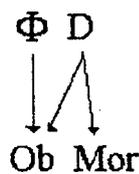


Figura 4.1: Sistema Formal Gerando Categoria

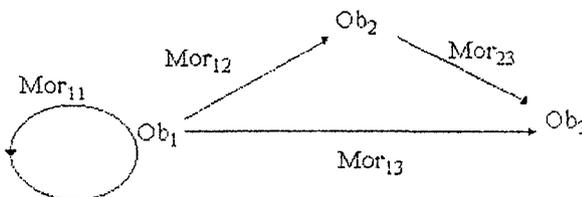


Figura 4.2: Representação dos objetos e morfismos numa Categoria

4.6.3 Representação de um Morfismo

Morfismos geralmente são representados por setas, iniciando no primeiro e terminando no segundo objeto do morfismo.

$$A \xrightarrow{f} B$$

Sobre a seta escreve-se o nome do morfismo.

4.6.4 Diagramas

A utilização da representação de morfismos por setas permite a construção de diagramas, que por sua vez permitem visualizar claramente as composições complexas de morfismos pois, o ser humano compreende melhor desenhos do que números em tabelas.

Definição 4.6.2 Diz-se que um diagrama é **comutativo** quando, em todo par de objetos, o uso de todo percurso indicado pelos morfismos produz o mesmo resultado.

4.6.5 Tipos de morfismos

A correspondência de objetos de um domínio em outro é produzida pelo morfismo, que preserva as características definidas em ambos. Por exemplo, no caso da Catego-

ria dos Conjuntos, os morfismos são funções. Os conjuntos podem ser manipulados dando outros conjuntos através de operações.

Monomorfismo

Se $fg = fh \Rightarrow g = h$, então, f é um monomorfismo conforme mostrado na Figura 4.3.

$$A \xrightarrow{f} B \begin{array}{c} \xrightarrow{g} \\ \xrightarrow{h} \end{array} C$$

Figura 4.3: Monomorfismo

Epimorfismo

Se o diagrama é comutativo, $gf = hf$. Se isto implica: $gf = hf \Rightarrow g = h$, então, f é um epimorfismo, conforme mostra a Figura 4.4.

$$A \begin{array}{c} \xrightarrow{g} \\ \xrightarrow{h} \end{array} B \xrightarrow{f} C$$

Figura 4.4: Epimorfismo

Isomorfismo

Conforme a Figura 4.5 o morfismo f é monomórfico se $g = h$. Da mesma forma f é epimórfico sendo $f = k$. Assim, f é isomórfico porque é monomórfico e epimórfico.

$$A \begin{array}{c} \xrightarrow{g} \\ \xrightarrow{h} \end{array} B \xrightarrow{f} C \begin{array}{c} \xrightarrow{l} \\ \xrightarrow{k} \end{array} D$$

Figura 4.5: Isomorfismo

4.6.6 Dualidade

Invertendo a flecha representativa do morfismo de uma dada categoria e obtém-se a sua co-categoria, ou categoria oposta!

4.6.7 Funtores

Outro conceito importante neste trabalho é a mudança de contexto, ou seja, mudança de categoria, que pode ser feita por um *Functor*. Esse functor é o que associa uma categoria a outra ou uma categoria a ela mesma. Por exemplo, para a ligação da Categoria dos Conjuntos com a Categoria dos Autômatos usa-se um functor.

Definição 4.6.3 Functor: *é o objeto matemático que, dadas duas categorias, associa objetos a objetos e morfismos a morfismos, e que satisfaz os seguintes condições:*

- Existir o functor identidade que associa uma categoria a ela mesma.
- Existir uma composição de funtores. Os funtores podem ser compostos gerando novos funtores.
- Existir uma composição associativa de funtores.

Nota-se que functor e morfismos são duas coisas bem distintas: um morfismo associa objetos de uma categoria e um functor associa duas categorias; um morfismo é representado por uma seta e um functor é representado por duas setas.

Os funtores podem ser monomórficos, epimórficos e isomórficos. Por exemplo, imagina-se o caso de duas categorias hipotéticas Cat_1 e Cat_2 e um functor F que liga objetos destas categorias. Se a cada par de objetos de Cat_2 é associado um único par de objetos de Cat_1 este functor é monomórfico (“mono” - só um). Se todos os pares de objetos de Cat_2 estão associados a Cat_1 este functor é epimórfico (“epi” - todos). Se o functor é monomórfico e epimórfico, então ele é isomórfico. Isto é, todos pares de objetos Cat_1 estão associados a Cat_2 uma única vez e não existe nenhum objeto de Cat_2 ao qual não se tenha uma associação.

Neste trabalho, o functor toma relevância por ser também a base de Engenharia de Programas, imaginando-se um programa como sendo um transformador (functor). Dado um conjunto de dados, estruturados, e as relações entre esses dados, pode-se definir uma categoria especificada de resultados. A formalização de um programa é, então, a formalização de um functor que vai associar dados a resultados.

Funtor Esquecedor

Um exemplo de funtor não epimórfico é o transformador de categorias na analogia de sistemas elétrico-mecânicos, pois existem sistemas mecânicos que não correspondem a nenhum sistema elétrico. Assim, no contexto mecânico, existiria elemento que não seria imagem de nenhum elemento do contexto elétrico, não existindo, portanto, funtor isomórfico.

Pode-se então utilizar um funtor esquecedor (“forget”), que desprezaria estes elementos indefinidos, produzindo-se a simplificação na transformação do contexto elétrico-mecânico. O funtor esquecedor faz aproximações desprezando determinadas condições.

Em termos categóricos, não existe funtor isomórfico na analogia de sistemas elétricos-químicos, pois nem tudo que existe no domínio elétrico existe no domínio químico. O domínio elétrico é mais potente do que o químico porque possui armazenadores de energia (capacitor e indutor) que não tem imagem no químico. Entretanto, tudo o que existe no domínio químico existe no análogo elétrico. Para a transformação químico-elétrico deve-se usar o funtor esquecedor porque o indutor, por exemplo, não é definido no análogo químico. Outras analogias, como químico-termodinâmico, também necessitam do funtor esquecedor.

4.6.8 Categoria dos Conjuntos

Para caracterizar uma categoria é preciso identificar os objetos e os morfismos. Supondo-se que sejam tomados por objetos os conjuntos, os morfismos terão que ser algo que associe conjunto a conjunto, havendo sentido falar em *primeiro* e *segundo* conjunto. Ora, um bom candidato a morfismo para a Categoria dos Conjuntos são as funções, porque uma função tem um conjunto para seu domínio e outro conjunto para seu contradomínio.

Então, define-se a Categoria dos Conjuntos que tem por objetos (*Ob*) os conjuntos, e por morfismos (*Mor*) as funções definidas entre conjunto domínio e conjunto contradomínio. Verifiquemos se isto satisfaz aos axiomas de uma categoria.

Definição 4.6.4 *Função: Associa, dado um conjunto, a outro conjunto ou a ele mesmo. Então existe, dado um morfismo (ou seja, dada uma função), um conjunto domínio e um conjunto contradomínio.*

Definição 4.6.5 *Composição de funções: As funções podem ser compostas formando uma nova função. Dadas duas funções f e g , existe uma função $h = f.g$, (isto é h é f composta com g).*

Definição 4.6.6 *Existe a função identidade; que é a função que manda um conjunto nele mesmo reproduzindo seus elementos. A composta de duas funções reproduz elementos ($h = f.g$) ou fazendo h (mesmo elemento), ou seja, existe tanto a possibilidade de calcular o valor da função f e depois da função h , quanto a de calcular a função g . Evidentemente, esta função é comutativa e existe a função identidade.*

Conseqüentemente, as funções são elementos passíveis de serem escolhidos como morfismos, na Categoria dos Conjuntos, por atenderem os requisitos da Teoria das Categorias:

- Morfismos se referem a pares de objetos;
- Possibilidade de composição associativa de morfismos;
- Existe o morfismo identidade.

Na categoria de conjuntos [3], identifica-se a seguir o que são monomorfismos, epimorfismos e isomorfismos:

Teorema 4.6.1 *Monomorfismos são funções injetoras.*

*Se $fg = fh \Rightarrow g = h$, então f é um monomorfismo.
 f, g, h são funções.*

Demonstração: Por absurdo:

Supondo-se que f não é injetora. Então existe b_1 e b_2 pertencente a B , existe ao menos um par b_1, b_2 no mesmo domínio (a imagem é mesma mas eles são diferentes). Isto é:

$$\exists!(b_1, b_2) \in B, b_1 \neq b_2 \text{ e } (b_1) = f(b_2)$$

Decorre que g e f podem ser diferentes. Se a função não for injetora pode haver a_1 (elemento de A) em que $h(a_1) = b_1$ e $g(a_1) = b_2$. Mas, $fg = fh$, portanto, f deve ser injetora.

Para ser injetora todo elemento de C é imagem só de um B . Não existem dois elementos diferentes de B que dão a mesma imagem em C . Isto obriga que g e h sejam iguais.

Teorema 4.6.2 *Epimorfismos são funções sobrejetoras. Nesta categoria de conjuntos, denominada ($Conj$), f, g, h são funções.*

Demonstração: Por absurdo:

Supondo-se que f não é sobrejetora. Se f não é sobrejetora, existe ao menos um elemento b que pertence a conjunto B , tal que b não é função de a pela função f , ou seja, nem todo elemento do contradomínio é imagem de todos os elementos. Isto é:

$$\exists! b \in B, b \notin f(A)$$

Existe um elemento de B que não é imagem de nenhum elemento de A pela função f . Assim, mesmo que $g(b)$ seja diferente de $h(b)$, terá imagem de b que não é imagem de nenhum elemento de g e h pois h e g são funções definidas em todo domínio de B . Então, os elementos b que não são imagens pela função f pertencem ao domínio de h e g . Observando-se a Figura 4.4 nota-se que, se $gf = hf \Rightarrow g = h$ então f é um epimorfismo, ou seja, todos os elementos de b são imagens de A .

Um exemplo, de interesse particular neste trabalho, é a categoria de conjuntos e sua relação com o Ensino: a categoria de conceitos a serem ensinados. Sua explicação é apresentada na seção 4.6.9.

4.6.9 Categoria de Conceitos

A Categoria dos Conceitos é um exemplo da Categoria dos Conjuntos na qual os objetos são conceitos e os morfismos são funções entre conceitos. O processo de Ensinar envolve duas categorias: uma categoria de conceitos a serem ensinados e uma categoria de conceitos existentes na mente do aprendiz.

Ensinar é modificar a categoria de conceitos existentes no aprendiz incluindo, na mesma, elementos vindos da categoria de conceitos a serem ensinados. Para que um elemento da categoria de conceitos a serem ensinados possa chegar até o aprendiz, é necessário passar por um meio (mídia) de comunicação. Deve-se considerar, então, que a categoria de conceitos a serem ensinados são limitados a mídia utilizada. Desta forma, a mídia tem um papel fundamental na eficácia da transmissão de conhecimentos, ou seja, na aprendizagem. Se for utilizada a linguagem falada, deve-se observar que nem sempre é possível exprimir o pensamento com palavras. Conseqüentemente, apenas uma visão parcial do pensamento pode ser verbalizável. Isto formalmente significa que a Categoria dos Conceitos a serem ensinados (ou transmitidos) é projetada na categoria de conceitos transmissíveis pela mídia vocal, a que realmente vai atingir o aprendiz. Assim, a Categoria dos Conceitos a serem ensinados é projetada na categoria da linguagem falada (mídia) e esta vai à categoria dos conceitos existentes no aprendiz através de um funtor que faz a simplificação daquilo que é falado. O ato de gerar frases representando o pensamento é materializado pelo funtor que traduz a categoria de conceitos a serem ensinados na

categoria das linguagens faladas para que os conceitos possam ser verbalizados como projeções do pensamento. Matematicamente, isto explica que: “jamais se transmite tudo o que se pensa”.

Por outro lado, os conceitos já existentes serão modificados com a adição de novos elementos da categoria de conceitos a serem ensinados, oriundos da categoria de conceitos transmissíveis pela mídia. Esses novos elementos que vão se juntar na categoria de conceitos já existentes, aumenta, assim, o número de elementos desta nova categoria. Novamente, através de um funtor inverso (processo de interpretação e compreensão), a categoria de conceitos existentes no aprendiz altera seu conteúdo. Quando não se entende uma frase isto significa, em termos formais, a incapacidade de fazer o funtor inverso. Se o funtor construído for muito diferente da imagem cognitiva que o aprendiz tem em seu cérebro, ele pode entender uma frase parecida ou não entender o que foi dito na frase³. Formalmente isto quer dizer: “nunca se entende completamente o que o outro diz”.

Então, do que foi exposto nesta abordagem categórica, no processo de ensino existe uma influência do meio que é utilizado. Conseqüentemente, se este meio for mais rico, a transmissão de conhecimentos deverá, em princípio, ocorrer de melhor maneira e isto é a justificativa precisa, matemática, da necessidade de utilização de tantos meios auxiliares (fala, música, animações, imagens, filmes, etc.) nos processos de Ensino-aprendizagem. O computador, sendo uma ferramenta que capaz de criar multimídias pode aumentar a eficácia na transmissão dos conhecimentos.

4.6.10 O Problema do Contexto na Categoria de Conceitos

Como saber se duas pessoas que estão conversando num dado momento, estão se entendendo? Geralmente, isto é feito detectando se as frases ditas estão dentro de um contexto comum, ou seja, é necessário verificar se os contextos da imagem cognitiva de cada um são semelhantes. Como perceber isso se as imagens cognitivas de cada pessoa não são acessíveis às outras? Uma pessoa pode perceber estas imagens cognitivas fazendo projeções⁴ das imagens cognitivas da outra.

Quando uma pessoa fala, ela não exprime todo seu pensamento porque a palavra é incapaz de exprimir a totalidade do pensamento, por este motivo muitas vezes usa-

³Isto ocorre quando existe ambigüidades, ou seja, quando dois pensamentos dão origem à mesma frase. Se dois pensamentos originam a mesma frase, na construção da imagem mental, pode-se obter um pensamento ou outro. Em linguagem categórica isto deve-se à construção de funtores que não são monomórficos. Os funtores monomórficos dariam uma representação única.

⁴Uma projeção é a simplificação daquilo que existe. Por exemplo, a projeção de um objeto de três dimensões em cima de um papel será o desenho deste objeto em duas dimensões.

se gestos para complementar as palavras. Dessa forma, tem-se então as projeções do pensamento. Quando duas pessoas conversando fazem projeções dos seus estados cognitivos e fazem crer que o estado cognitivo de ambas é muito semelhante, diz-se que estas pessoas estão se entendendo. Caso contrário, se o estado cognitivo entre estas pessoas é totalmente diferente, elas fazem o que se chama comumente de “conversa de surdos” (uma não “ouve” a outra). Por exemplo, supondo-se a seguinte conversa entre duas pessoas (P1 e P2) do exemplo 4.6.1:

Exemplo 4.6.1 Estados cognitivos diferentes:

P1: Hoje vamos ter sol e calor o dia inteiro.
P2: Mas acontece que meu time ganhou ontem.
P1: Recebi uma carta da minha tia.
etc...

Chega-se à conclusão de que os estados cognitivos são bem diferentes pois, P1 fala no contexto de clima, P2 responde no contexto do futebol e, em seguida, P1 muda para outro contexto falando da carta de sua tia.

Muito diferente do diálogo telefônico, por exemplo (no qual as pessoas se entendem) do exemplo 4.6.2:

Exemplo 4.6.2 Estados cognitivos semelhantes:

P1: Hoje vamos ter sol e calor o dia inteiro.
P2: Então vamos aproveitar para ir à praia.
P1: Certo, então vamos para a Tapera porque lá não tem ondas fortes.
P2: Ótimo! Então passa aqui que já vou colocar a roupa de banho...
etc...

Neste caso, o contexto é o mesmo. Os estados cognitivos das pessoas são semelhantes. Após P1 dizer que o dia estava ensolarado e quente, P2 sugere que fossem à praia. Apesar de que P1 ter feito uma simplificação, não mencionando que “Tapera” trata-se do nome de uma praia⁵, P2 entendeu que poderiam ir na praia da Tapera pois isto estava dentro do contexto. Se P2 não soubesse o que é Tapera, e que ondas fortes são fenômenos naturais encontradas em muitas praias, certamente perguntaria coisa do tipo “O que é Tapera? Onde fica?” e não falaria “o Cruzeiro é o melhor time do Brasil”.

⁵A palavra Tapera origina-se do Tupi e também pode significar casa, aldeia ou fazenda abandonada além de outros significados que podem ser encontrados em [79].

Quando se começa uma conversação, existe um processo de sintonia de estados cognitivos, no qual o estado cognitivo de uma pessoa é alterado para sintonizar com estado do meio. Geralmente, através de perguntas sintoniza-se no estado cognitivo do outro porque este não sabe qual é o estado cognitivo do seu interlocutor e nem sua capacidade de desenvolvê-lo. Quando não é possível seguir o estado cognitivo do outro, as pessoas fazem perguntas que são projeções de seu próprio estado. A projeção do estado cognitivo de uma pessoa provoca a modificação do estado da outra para que a mesma possa associar o seu estado cognitivo com o estado do outro.

Para que haja aprendizado é necessário uma sintonia de estados cognitivos [222]. Esta sintonia deve ser tal que permita uma transmissão de conhecimentos⁶.

A construção de frases é uma tentativa de fazer com que haja projeções do estado cognitivo da parte que não existe no estado cognitivo do outro, através de um meio de transmissão de conhecimentos, ou seja da verbalização. Verbalizar é algo muito difícil, mas é um processo de projeção de um estado cognitivo. Para efetuar estas projeções de pensamento é necessário que as pessoas façam seleções (por exemplo, do que vai dizer) e construir modelos. Estes modelos é o que se chama de contexto. No exemplo 4.6.1, P1 fala do clima e P2 fala de futebol, contextos diferentes. Assim, se os contextos mudam, os estados cognitivos mudam, torna-se necessário, para que duas pessoas se entendam, como no exemplo 4.6.2, é a fala dentro do mesmo contexto e, dessa forma seus estados cognitivos sejam semelhantes e haja sintonização.

O contexto é a base do estado cognitivo. Então, este estado cognitivo desenvolve-se dentro de contextos. Observando um exemplo do uso da palavra “manga” no contexto das frutas, conforme o exemplo 4.6.3, e outro no contexto das roupas, como o exemplo 4.6.4, tem-se:

Exemplo 4.6.3 Uso da palavra “manga” - Diálogo numa feira:

P1: - Esta manga está boa?

P2: - Uma delícia... vai levar uma?

P1: - Não, muito obrigada, prefiro levar abacate.

Exemplo 4.6.4 Uso da palavra “manga” - Diálogo numa alfataiaria:

P1: - Esta manga está boa?

P2: - Acho que você pode diminuir um pouco mais.

P1: - Mas não vou diminuir demais porque não se usa mais mangas de camisas tão curtas.

⁶O conhecimento seria então dizer que existiria no estado cognitivo de uma pessoa mais coisas sobre um determinado assunto do que existe no estado cognitivo da outra.

Observando o exemplo 4.6.3, o diálogo hipotético pode ser entre duas pessoas numa feira e, o do exemplo 4.6.4 pode se passar numa alfaiataria. Nestes casos, o contexto é muito importante para eliminar ambigüidades. O problema todo da comunicação humana é o da definição de contextos.

No caso, do Ensino uma aplicação é saber se um curso está adaptado ou não ao aluno. Se não estiver adaptado ao aluno significa que o estado cognitivo necessário para a interpretação do conteúdo não é compatível com o estado cognitivo inicial dos alunos, ou seja, é uma maneira formal de dizer que “o curso está difícil demais”. Uma solução é o ensinamento de conceitos, partindo-se dos mais simples aos mais complexos, aplicando-se na categoria que define o contexto da matéria, um functor denominado “esquecedor”, para produzir uma outra categoria mais simples, tendo-se, assim, uma nova categoria simplificada, ou seja, só de conceitos que sejam capazes de ser transmitidos. De certa forma, isto é formalizar (através da Teoria das Categorias) a Teoria de Piaget e o Construtivismo.

4.7 Outras Categorias

A Teoria da Categoria dos Conjuntos ensina a raciocinar com funções em lugar de elementos. As demonstrações feitas em termos de elementos são válidas nesta categoria denominada “Conj”, mas não são válidas em outras categorias, nas quais o raciocínio primitivo deve ser feito baseado em morfismos.

Uma categoria é construída de acordo com o referido contexto. No caso da Categoria de Conjuntos, as funções foram escolhidas como morfismos, mas também é possível definir outros morfismos.

Exemplo 4.7.1 *Exemplo de categorias:*

- *Categoria dos Grafos - Objetos: Grafos, Morfismos: percursos;*
- *Categoria dos Nós de Grafos - Objetos: conjunto de nós, Morfismos: conjunto de arestas;*
- *Categoria de Estados do Autômato - Objetos: conjunto de Estados, Morfismos: dinâmica do Auômata.*
- *etc...*

Um resumo da Teoria das Categorias pode ser encontrado nos “Dogmas de Go-guen” [98]:

1. “To each species of mathematical structure, there corresponds a **category** whose objects have that structure, and whose morphisms preserve it.”
2. “To any natural construction on structures of one species, yielding structures of another species, there corresponds a **functor** from the category of the first species to the category of the second.”
3. “To each natural translation from a construction $F : A \rightarrow B$ to a construction $G : A \rightarrow B$ there corresponds a **natural transformation** $F \Rightarrow G$.”
4. “A **diagram** D in a Category C can be seen as a system of constraints, and then a **limit** of D represents all possible solutions of the system.”
5. “To any canonical construction from one species of structure to another corresponds an **adjunction** between the corresponding categories.
6. “Given a species of structure, say widgets, then the result of interconnecting a system of widgets to form a super-widget corresponds to taking the **colimit** of the diagram of widgets in which the morphisms show how they are interconnected.”
7. “Given a species of structure C , then a species of structure obtained by “decorating” or “enriching” that of C corresponds to a **comma category** under C (or under a functor from C ”).

Capítulo 5

Hipertômatos

“See simplicity in the complicated.”

Lao Tzu.

Neste capítulo serão apresentados os conceitos básicos da Teoria dos Autômatos e de Sistemas Hipertexto (ou hipermídia) como Autômato, visando mostrar o ferramental teórico necessário na unificação destas duas teorias à Teoria das Categorias. O capítulo também apresenta as implementações desenvolvidas seguindo o modelo Hipertômato cujos conceitos introdutórios foram apresentados nos Capítulos 4 e 5 com a intenção de mostrar a potencialidade da formalização de sistemas em termos práticos.

5.1 Definição teórica de Hipertexto

Os trabalhos de Pagano [165], [166], [162], [164] e [163] apresentam uma definição teórica de hipertexto. Segundo Pagano, o hipertexto tem sido apresentado dentro da perspectiva teórica como um grafo, como uma rede semântica ou como rede de Petri. A maioria destas abordagens não incorpora todas as características do hipertexto sendo formalismos de natureza declarativa que não consideram os aspectos dinâmicos do hipertexto. Um modelo formal de hipertexto ou de hipermídia (o computador gerencia vários meios para apresentar a informação e isto justifica porque as implementações de hipertexto são, às vezes, denominadas de hipermídia), pode ser fundamentado na Teoria de Autômatos, incorporando não somente os aspectos declarativos do sistema (por exemplo, o diagrama de transição), mas também

os aspectos dinâmicos (a evolução no tempo do autômato em resposta a qualquer seqüência de entrada de dados).

5.2 Conceitos básicos da Teoria dos Autômatos

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 “causal”: 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. Em um sistema dinâmico descreve-se um sistema como se descrevesse o mecanismo de como ele trabalha (internamente), especificando como o conjunto dos estados varia com o tempo [33]. Tal descrição é suficiente para gerar uma definição comportamental.

Formalmente nós temos:

Definição 5.2.1 *Um Sistema Dinâmico é o objeto matemático descrito como:*

$$S_d = \{T, U, \Psi, Y, \Lambda, X, \Sigma, \delta\}$$

onde:

T é o conjunto dos tempos,

Ψ é o conjunto de funções de entrada $\psi \in \Psi = \{\psi : T \rightarrow U\}$,

U é o conjunto dos valores de entradas,

Y é o conjunto dos valores de saídas,

Λ é o conjunto de funções de saída $\lambda \in \Lambda = \{\lambda : T \rightarrow Y\}$,

X é o conjunto dos estados,

Σ é a função de transição de estados $\Sigma : T \times T \times \Psi \rightarrow X$,

δ é a função de saída $\delta : T \times X \times U \rightarrow Y$,

Informalmente, a noção de sistema dinâmico corresponde a um sistema funcional temporal cujo estado varia com o tempo dependendo do valor da entrada. Assim, todos os sistemas dinâmicos são sistemas temporais e funcionais. As escolhas particulares dos conjuntos envolvidos na definição de sistema dinâmico conduzem a diferentes tipos de sistemas. Detalhes sobre tais sistemas podem ser obtidos em Barreto [33].

Definição 5.2.2 *Num sistema temporal Ψ e Λ são funções do tipo:*

$$\Psi : T \rightarrow U \text{ e } \Lambda : T \rightarrow Y$$

onde:

U é o conjunto dos valores da entrada,

Y é o conjunto dos valores da saída,

T é um conjunto orientado com um primeiro elemento, comumente denotado por t_0 (às vezes é usual considerar t_0 como $-\infty$), que é chamado de conjunto de tempo T .

O conceito de sistema como uma relação matemática não è, às vezes, rico o suficiente para descrever o sistema, para estudá-lo em um nível suficiente de detalhes. Frequentemente, deseja-se ter uma única saída para uma única entrada. Em uma relação pode haver muitos valores de saída correspondentes a um valor de entrada. Isto leva ao conceito de:

Definição 5.2.3 Um sistema funcional é caracterizado por uma função f :

$$f : \Psi \times X \rightarrow \Lambda$$

onde:

Ψ : conjunto das entradas admissíveis,

Λ : conjunto das saídas admissíveis,

X : conjunto dos estados.

Definição 5.2.4 Um Sistema Dinâmico Contínuo no Tempo (ou sistema contínuo no tempo) é um sistema dinâmico onde:

T é um subconjunto dos números reais,

X, U, Y são subconjuntos de $\mathbb{R}^n, \mathbb{R}^m, \mathbb{R}^p$, espaço real n, m, p -dimensional,

Σ é um conjunto de funções diferenciáveis em relação a t .

Definição 5.2.5 Um Sistema Dinâmico Discreto no Tempo (ou sistema discreto no tempo) é um sistema dinâmico no qual o conjunto dos tempos é um subconjunto dos inteiros.

Definição 5.2.6 Um Sistema invariante no tempo é um sistema dinâmico cuja função de transição Σ depende de um único elemento de T e a função de saída é independente de T .

É útil tomar como uma primeira aproximação de um sistema real um modelo invariante no tempo ou em sistema estacionário. O fato de que a função de transição depende somente de um elemento de T , significa que o valor do estado não depende do tempo inicial nem do tempo considerado, mas somente do intervalo de tempo entre o tempo inicial e o tempo considerado. A função de saída sendo independente do tempo significa que, para qualquer instante, estado e entrada iguais, produz-se a mesma saída. Se este é o caso, para qualquer tempo inicial, somente a duração do experimento é importante para determinar o estado e a saída em qualquer instante.

Como não é muito fácil lidar com sistemas variantes no tempo, às vezes, o espaço de estado é enriquecido com uma nova variável tempo e o sistema resultante não é mais um sistema variante no tempo. Então, pode-se dizer que a qualquer sistema variante no tempo corresponde um sistema invariante no tempo, no qual o tempo foi considerado como um novo elemento do conjunto dos estados, este o conjunto dos tempos¹.

Um tipo muito usual de sistema dinâmico em Ciência da Computação é a *máquina de estados finitos*. Informalmente, uma máquina de estados finitos é um sistema dinâmico onde o conjunto dos tempos é o conjunto dos inteiros, e a entrada, a saída e os estados são conjuntos finitos. Neste caso, os valores possíveis da entrada e da saída são referidos como alfabetos de entrada e saída. Sob estas simplificações, não é essencial indicar explicitamente o conjunto dos tempos T nem explicitamente introduzir Ψ e Λ . Por outro lado, é usual declarar explicitamente um estado inicial correspondente ao tempo zero [33].

Definição 5.2.7 *Um autômato (ou máquina) é descrito abstratamente como uma sêxtupla [33]:*

$$A_t = \{U, Y, X, x_0, \lambda, \eta\}$$

onde:

U é um conjunto finito de entradas,

Y é um conjunto finito de saídas,

X é um conjunto de estados ou espaço de estado,

$x_0 \in X$ é o estado inicial,

$\lambda : U \times X \rightarrow X$ é a função de próximo estado ou função de transição,

$\eta : U \times X \rightarrow Y$ é a função de próxima saída.

¹Esta equivalência entre sistemas variantes e invariantes no tempo pode parecer muito abstrata. Contudo, na escrita do código de simulação para sistemas variantes no tempo, ela é frequentemente utilizada [33].

Definição 5.2.8 *Um autômato é um sistema dinâmico invariante e discreto no tempo. Quando o espaço de estado é um conjunto finito, o autômato é chamado de autômato finito .*

Então, esta sêxtupla formal é interpretada como sendo uma descrição matemática de uma máquina à qual, se no tempo t_0 estiver no estado x_0 e receber um segmento de entrada u do tempo t_0 ao tempo t , estará no tempo t no estado $\lambda(x, u)$ e emitirá a saída $\eta(x, u)$ [33].

Exemplo 5.2.1 *Considerando-se o autômato M :*

$$M = \{U, Y, X, x_0, \rho, \delta\}$$

onde:

$U = \{0, 1\}$, conjunto de entradas,

$Y = \{\alpha, \beta, \pi\}$, conjunto de saídas,

$X = \{A, B, C\}$, conjunto de estados,

x_0 , estado inicial que pode ser tanto A , B como C ,

$\rho : U \times X \rightarrow X$, função de próximo estado,

$\delta : U \times X \rightarrow Y$, função de próxima saída.

Os valores das funções ρ e δ são dados por:

$$\rho(0, A) = C \quad ; \quad \delta(0, A) = \beta$$

$$\rho(1, A) = A \quad ; \quad \delta(1, A) = \alpha$$

$$\rho(0, B) = A \quad ; \quad \delta(0, B) = \pi$$

$$\rho(1, B) = C \quad ; \quad \delta(1, B) = \beta$$

$$\rho(0, C) = B \quad ; \quad \delta(0, C) = \alpha$$

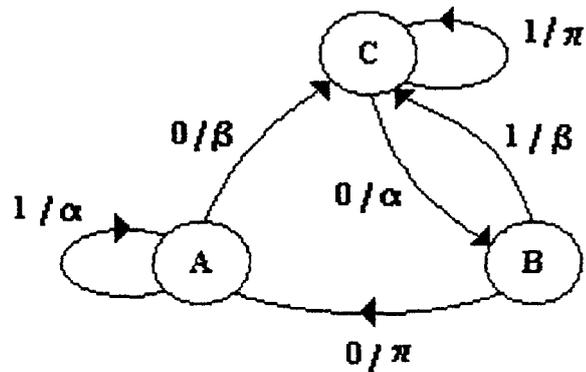
$$\rho(1, C) = C \quad ; \quad \delta(1, C) = \pi$$

Pode-se construir a Tabela 5.1 de transição de estados ρ para autômato M .

Este autômato M pode ser representado por um grafo orientado conforme ilustra a Figura 5.1. Nota-se que o mesmo estado de um autômato pode ter uma ou mais saídas. Por exemplo, o estado A do autômato M pode ter as saídas α ou β . E ainda, dois estados diferentes, como A e B do autômato M podem ter a mesma saída, como por exemplo, β .

Essencialmente existem dois tipos de representação em grafos.

Estado presente X	A	A	B	B	C	C
Entrada U	0	1	0	1	0	1
Próximo estado Y	C	A	A	C	B	C
Próxima saída δ	β	α	π	β	α	π

Tabela 5.1: Transição de estados do autômato M Figura 5.1: Grafo representando o autômato M

1. A cada estado é associado um nó do grafo (círculo) e para cada transição de estado associa-se os arcos (com setas). Ao lado de cada arco orientado (setas), são indicados: as entradas aplicadas ao estado original (início da seta) e a saída resultante no próximo estado (final da seta).
2. Associa-se a cada subconjunto de nós um estado. A função de saída tem por argumentos este subconjunto de nós e a entrada que faz o mapeamento na saída, que persiste, até a próxima entrada.

Definição 5.2.9 *Dois autômatos que apresentam o mesmo par de segmento de entrada e segmento de saída são ditos equivalentes no nível comportamental.*

5.3 Hipertexto como autômato

A introdução de características dinâmicas parece aumentar as possibilidades do hipertexto (ou hiperfídia) em ambientes de aprendizagem computadorizados, conforme demonstrados em Pagano [165], [166], [162], [163]. Um modelo teórico de

hipertexto necessita de três conceitos a serem incorporados: o nó de informação, a ligação entre os nós e os nós de simultaneidade, sendo que este último conceito, considera que um ou mais nós podem ser apresentados simultaneamente na tela.

Um sistema computacional, denominado hipertexto, pode ser convenientemente definido como autômato [169]:

Definição 5.3.1 *Um hipertexto H_p é descrito abstratamente como a séxtupla:*

$$H_p = \{U, Y, X, x_0, \rho, \delta\}$$

onde:

U é o alfabeto finito de entrada,

Y é o alfabeto finito de saída,

X é o espaço de estado finito,

$x_0 \in X_0 \subset X$ é o estado inicial,

$\rho: U \times X \rightarrow X$ é a função de transição de estados,

$\delta: U \times X \rightarrow Y$ é a função de saída do próximo estado.

O alfabeto de entrada U do hipertexto H_p é o conjunto de valores que o usuário pode introduzir no sistema através do teclado, “mouse” ou qualquer outro dispositivo. As ações como mudança de tamanho de uma janela, acionamento de botão, etc., geram o alfabeto de entrada.

O conceito de *estado* conduz ao aspecto dinâmico do hipertexto. A possibilidade de apresentação simultânea de nós de um hipertexto é essencialmente uma porção deste modelo através do conceito de estado. Se cada nó de informação é associado, por exemplo, com uma janela na tela do computador (uma forma possível de saída), então, um conjunto de janelas na tela caracteriza um estado x do hipertexto H_p .

O estado inicial $x_0 \in X$ do hipertexto H_p é qualquer estado no qual o usuário pode começar a navegar no documento. Pode existir um ou mais estados iniciais (configuração de nós) dos quais, por exemplo, um deles pode iniciar a localização de uma base de dados ou iniciar a navegação. Neste caso, tem-se um conjunto de todos os estados iniciais possíveis X_0 que é um subconjunto de todos os estados possíveis, ou seja, $x_0 \in X_0 \subset X$.

A função de transição pode ser referida como uma ligação (“link”) que “junta” o hipertexto como um todo.

A função de transição ρ do hipertexto H_p é a função que conduz o sistema hipertexto de um estado a outro, segundo uma seqüência de dados da entrada.

O usuário poderá notar a transição de estado a outro se o conjunto de nós apresentados a ele em um novo estado tiver valores de saída diferentes do estado anterior.

O alfabeto de saída Y de um hipertexto H_p inclui todas as formas de representação da informação que são vistas pelo usuário na tela. O tamanho e a forma de cada janela na tela, suas posições relativas, gráficos, sons, vídeo, etc., caracterizam um alfabeto de saída.

A função de transição δ de um hipertexto H_p é uma função que, dada uma seqüência de entrada e um estado, oferece a informação e sua representação na tela ou em qualquer outro dispositivo. As possibilidades de apresentação dependerão da riqueza do alfabeto de saída. A função δ possui valores no alfabeto de saída, um conjunto de meios de apresentação que depende do estado dado e do alfabeto de entrada.

Se cada nó de informação é associado, por exemplo, com uma janela na tela do computador (uma forma possível de entrada), então um conjunto de janelas na tela caracteriza um estado x do hipertexto. Pagano [169] colocou isso em termos formais da seguinte forma:

Considerando n como um nó, isto é, a unidade indivisível da informação, sendo N o conjunto de todos os nós existentes no documento:

$$N = \{n_1, n_2, \dots, n_p\}$$

Sendo $P(N)$ o conjunto de partes de N , então:

$$P(N) = \{\phi, \{n_1\}, \{n_1, n_2\}, \dots, N\}$$

O conjunto de estados ou espaço de estados X do hipertexto H_p é igual a $P(N)$. O estado $x \in X$ do hipertexto H_p é um elemento de $P(N)$.

De modo a ilustrar o conceito de hipertexto como um autômato são consideradas duas situações - dois elementos diferentes no conjunto de tempos - que serão descritos a seguir e correspondem à Figura 5.2. Este exemplo ilustra também a segunda representação dos grafos conforme mostrado na subseção 5.2.

Exemplo 5.3.1 *No lado esquerdo da Figura 5.2 são mostrados os nós 20 e 30 de um hipertexto qualquer. Uma entrada do tipo "acionar o mouse no nó 30" torna a saída conforme mostrado no lado direito da Figura 5.2. Observa-se que a saída mudou mas o estado não.*

No Exemplo 5.3.1, para cada valor de tempo corresponde um e somente um estado. Nesta situação, o estado do hipertexto H_p , em um determinado instante, é $x_1 = \{n_{20}, n_{30}\}$ e o alfabeto de saída é y_1 . Isto é, y_1 é a apresentação visual, o conjunto de janelas, como visto na tela esquerda da Figura 5.2 (gráficos, tamanho, posição relativa, etc.).

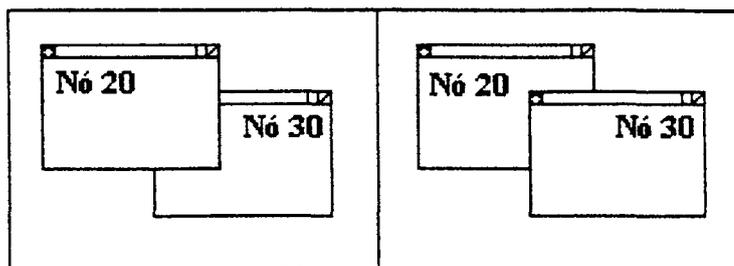


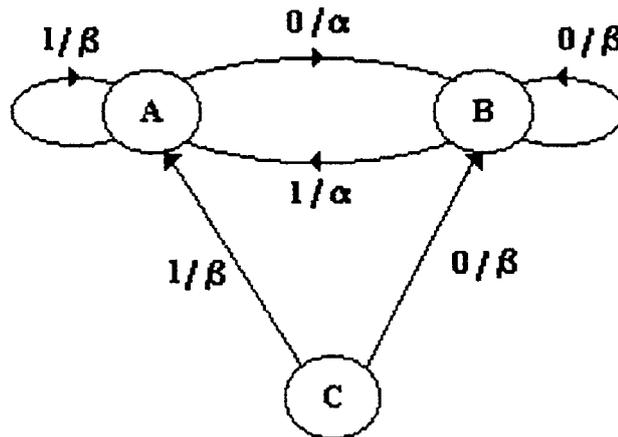
Figura 5.2: Mudança de saída e inalteração de estado no hipertexto H_p

A saída y_1 permanece até o tempo no qual a entrada u_1 = “acionar o mouse no nó 30” produz a saída y_2 , cuja situação é dada no lado direito da Figura 5.2, onde a janela 30 move-se para frente. Isto significa que para o estado x_1 e a entrada u_1 a função de saída δ conduz à saída y_2 (nota-se que a posição relativa das janelas mudou). Todavia, neste caso, quando a entrada u_1 foi aplicada no estado x_1 , a função de transição leva ao mesmo estado $x_1 = \{n_{20}, n_{30}\}$. Pode-se atingir o mesmo estado com uma saída diferente porque a função de saída depende da entrada e do estado ($\delta : U \times X \rightarrow Y$).

5.4 Estados observáveis e estados alcançáveis

De acordo com Pagano [169], dois problemas podem surgir na construção e interconexão de documentos em um hipertexto. O primeiro é assegurar que a unidade de informação seja observável, isto é, que se possa determiná-la pela observação do comportamento entrada/saída do sistema. O outro problema é assegurar que a unidade do sistema seja alcançável, ou seja, acessível pelo usuário, dado seu estado inicial (ponto de partida do usuário na leitura do documento).

Para que um estado x do hipertexto seja observável, o usuário deverá navegar os estados de um hipertexto através do desenvolvimento de experimentos simples e perceber a transição de estado sobre o par entrada/saída. Um experimento simples poderia ser a entrada de um segmento de dados, através de qualquer dispositivo de entrada, e a observação da saída do segmento de dados, através de qualquer dispositivo de saída. Para dizer que todos os estados de um hipertexto são alcançáveis é necessário que o usuário seja capaz de navegar através de todos os estados do hipertexto. A solução é ter um nó mapa na implementação para oferecer uma visão geral do sistema.

Figura 5.3: Representação de um hipertexto H

Exemplo 5.4.1 Considerar o hipertexto H representado na Figura 5.4 com o alfabeto de entrada $U = \{0, 1\}$, o alfabeto de saída $Y = \{\alpha, \beta\}$ e o conjunto de estados $X = \{x_1 = A, x_2 = B, x_3 = C\}$.

Se $x_0 = B$ é o estado inicial, não existe valor de entrada u para o qual, partindo-se do estado $x_0 = B$, possa se atingir o estado $x_2 = C$. Então, diz-se que o estado x_2 é **inalcançável**. Também examinando-se o hipertexto H são encontrados estados não observáveis. Por exemplo, o estado $x_1 = A$ é não observável porque dado o par de entrada-saída $(u = 1, y = \beta)$ não se sabe se o estado anterior era x_1 ou x_2 . Isto é, a entrada $u = 1$ no estado $x_1 = A$ conduz a saída $y = \beta$, a mesma saída é obtida para a entrada $u = 1$ no estado $x_2 = C$.

Estes conceitos são importantes no sentido de que, se dois hipertextos são equivalentes, o usuário não pode fazer distinção entre eles, exceto em detalhes computacionais. Ao se conectar nós de informação é desejável ter um conjunto mínimo de ligações para que todos os estados sejam alcançáveis e observáveis. O grafo de implementação permite a verificação de que todos os estados do sistema sejam alcançáveis e observáveis, evitando-se, assim, os problemas de navegação.

5.5 Definição de Hipertômato

O modelo de hipertexto como autômato desenvolvido por Pagano [169] foi ampliado por Almeida [5] e denominado *Hipertômato*. O modelo ampliado inclui os recursos hipermídia e sua utilização em redes de computadores na concepção de vários am-

bientes de Ensino. Os grafos de implementações podem mudar conforme o ambiente a ser concebido. Assim, tem-se a definição do Hipertômato em concordância com a definição do modelo de Pagano conforme apresentado na definição 5.3.1.

Definição 5.5.1 *O sistema formal, denominado Hipertômato H_t , é descrito como uma sextupla:*

$$H_t = \{U, Y, X, x_0, \rho, \delta\}$$

onde:

U é o alfabeto finito de entrada;

Y é o alfabeto finito de saída;

X é o espaço finito de estados;

$x_0 \in X_0 \subset X$ é o estado inicial;

$\rho : U \times X \rightarrow X$ é a função de transição de estados;

$\delta : U \times X \rightarrow Y$ é a função de saída do próximo estado.

Todas as considerações feitas na seção 5.3 são válidas para os modelos Hipertômatos. Para uma melhor compreensão da história desses modelos, a Figura 5.4 mostra a evolução no tempo dos trabalhos desenvolvidos por Pagano [169] e Almeida [5], [6]. Pagano iniciou seus trabalhos nos anos 80 e defendeu sua Tese de Doutorado na Faculdade de Ciências Aplicadas da Universidade de “Louvain la Neuve”, na Bélgica, em 1992. Almeida, dando prosseguimento aos trabalhos de Pagano, incluiu na modelagem as características da Hipermídia no Ensino via redes de computadores e sua unificação com a teoria das Categorias (1998-2002).

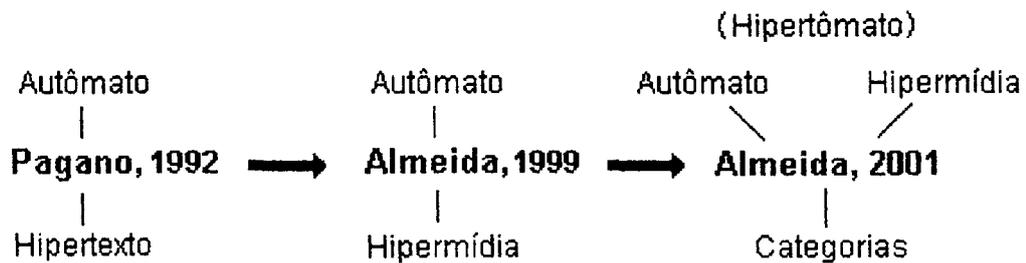


Figura 5.4: Evolução dos modelos de Pagano e Almeida

5.6 Categoria dos Autômatos

Na Categoria dos Autômatos [1], [211], os objetos são Autômatos e os morfismos podem ser escolhidos dentre as coisas que permitem associá-los. Pode-se identificar percursos como candidatos a morfismos quando estes atenderem às seguintes condições:

- Existir o Percurso Identidade - Dado um percurso, existe um outro que retorna a seu ponto de partida;
- Existir o Percurso Composto Associativo - Dados dois percursos pode ser obtido um terceiro pode ser obtido pela composição dos dois primeiros ou pela associação de percursos;

Na Categoria dos Estados de um Autômato os morfismos podem ser a dinâmica deste Autômato, ou seja, a mudança de estados. Assim, pelas condições categóricas para que a dinâmica do Autômato seja escolhida como um morfismo deve:

- Existir a Dinâmica identidade (continuar no mesmo estado);
- Existir a Dinâmica composta associativa (ir de um estado a outro e depois a um terceiro estado ou ir diretamente do primeiro para um terceiro estado);

A potencialização categórica não é única. É a que for mais conveniente, desde que satisfaça as condições para ser uma categoria. A Teoria das Categorias serve para simplificar o raciocínio unificando teorias através de funtores podendo, assim, traduzir resultados de um domínio para o outro. Esta Teoria não oferece nenhuma ferramenta de cálculo além das já existentes, mas oferece de maneira resumida de teorias, a compreensão [171] de fatos e uma linguagem muito potente de unir conceitos.

A Figura 5.5 ilustra esta caracterização em um modelo generalizado.

Este modelo geral representa os sistemas hipermídia como autômatos formando a Categoria de Sistemas Hipermídia como Autômato, ou seja, a Categoria de Hipertômatos .

5.7 Categoria de Estados de um Hipertômato

Definição 5.7.1 *A Categoria de Estados de um Hipertômato Cat_H , é o par (Ob, Mor) onde os objetos Ob são os estados do Hipertômato e os morfismos Mor são funções de transição de estados (ou simplesmente, percursos) no Hipertômato. Para satisfazer as condições categóricas deverá existir o morfismo identidade e uma composição associativa de morfismos.*

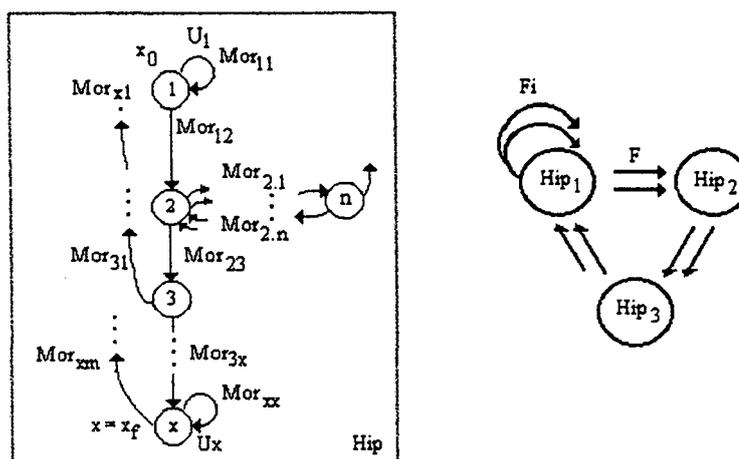


Figura 5.5: Modelo generalizado: Categoria de Hipertômatos

Exemplo 5.7.1 Dados 3 estados x_1, x_2 e x_3 de um Hipertômato H_i , este pode ser categorizado se satisfizer às seguintes condições:

- Existir o morfismo identidade Mor_{11} , ou seja, a transição não provocar uma mudança de estado.
- Existir uma composição associativa de morfismos:

$$Mor_{12} \circ Mor_{23} = Mor_{13}$$

A Figura 5.6 mostra que para ir ao estado x_3 é possível compor as transições (ou percursos no Hipertômato), de forma que, do estado x_1 à transição (Mor_{12}) conduz ao estado x_2 e a transição Mor_{23} conduz ao estado x_3 . O mesmo percurso pode ser feito diretamente do estado x_1 ao estado x_3 através do morfismo M_{13} .

5.8 Categoria dos Hipertômatos

Os Hipertômatos podem ser associados em uma Categoria dos Hipertômatos conforme mostra a Figura 5.7. O funtor associa os objetos (estados) de um Hipertômato a outro e morfismos a morfismos (transições ou percursos), satisfazendo às seguintes condições:

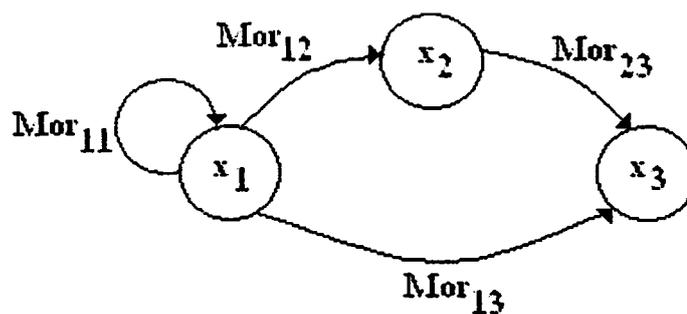


Figura 5.6: Categoria de Estados de um Hipertômato

- Existência do Funtor Identidade F_I que associe uma categoria de Hipertômatos a ela mesma;
- Composição dos Funtores de maneira associativa.

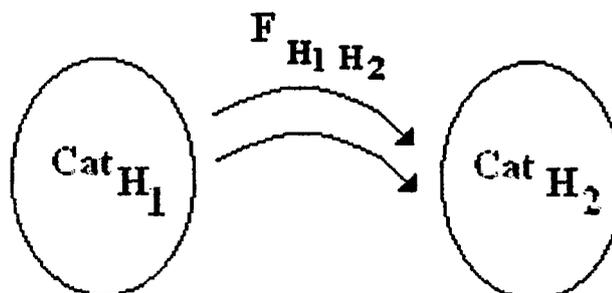


Figura 5.7: Categoria de Hipertômatos

Definição 5.8.1 *Funtor Monomórfico F_m* : Dadas as categorias de Hipertômatos Cat_{H_1} e Cat_{H_2} , a cada par de estados de Cat_{H_2} é associado um único par de estados de Cat_{H_1} .

Definição 5.8.2 *Funtor Epimórfico F_E* : Considerando as categorias Cat_{H_1} e Cat_{H_2} , todos os pares de estados de Cat_{H_1} estarão associados aos pares de estados de Cat_{H_2} .

Definição 5.8.3 *Funtor Isomórfico F_s* : Considerando as categorias Cat_{H_1} e Cat_{H_2} , todos os pares de Cat_{H_1} estão associados a Cat_{H_2} uma única vez e não existe nenhum objeto de Cat_{H_2} ao qual não se tenha uma associação.

A Figura 5.8 mostra um Funtor Isomórfico - monomórfico e epimórfico ao mesmo tempo.

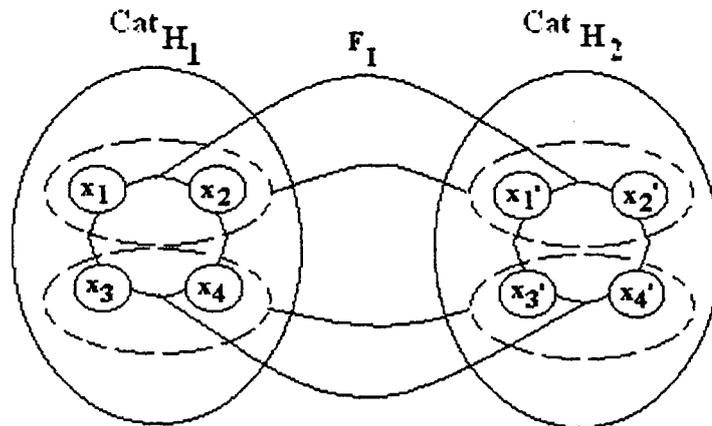


Figura 5.8: Funtor Isomórfico associando Hipertômatos

5.9 Modelagem categórica: um novo caminho para um velho problema

Existe uma grande preocupação no desenvolvimento de sistemas hipermídia com os recursos que a ferramenta de autoria oferece. Pouca ênfase é dada à concepção do sistema. Entretanto, um projeto mal feito pode destruir todo o esforço de enriquecimento da mídia. Em linguagem categórica, a existência de não monomorfismos pode conduzir a erros de navegação.

Um exemplo que ilustra isso muito bem: muitas vezes têm-se páginas na WWW com recursos visuais interessantes, mas ... de repente, ao acessar outra página o usuário não consegue retornar ao ponto em que estava e outras vezes encontra mensagens de erro do tipo "página não encontrada", conforme mostra a Figura 5.9.

No caso dos sistemas dedicados ao Ensino [84], os problemas de navegação podem trazer conseqüências tão desastrosas para o processo de aprendizagem quanto às produzidas num livro convencional com muitas páginas arrancadas. O computador deixa, neste caso, de ser uma ferramenta estimulante para tornar algo irritante e difícil de se lidar.

A navegação é um problema prático de sistemas hipertextos. Não adianta a colocação de recursos atrativos, sons, filmes e animações, (que certamente colabo-

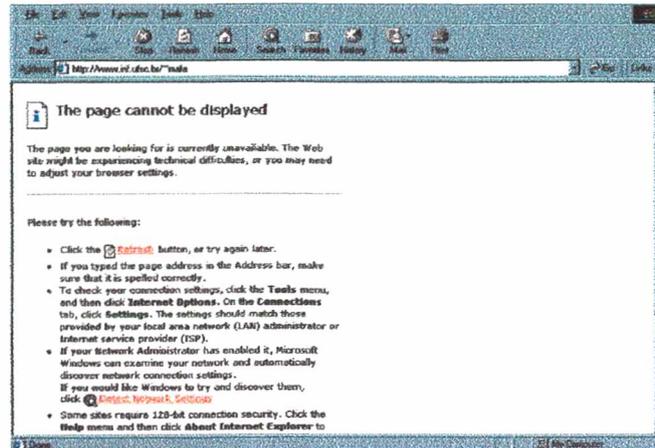


Figura 5.9: Exemplo de não monomorfismo na navegação

ram com o processo de Ensino-aprendizagem) [133], [140], [197], se estes recursos não forem mediados por uma concepção coerente. Infelizmente, apesar de toda a explosão da rede mundial [208] e das aplicações que são desenvolvidas com as modernas ferramentas de autoria multimídia (e hipermídia), nota-se que não existe muita preocupação com a especificação formal destes sistemas. Uma tendência geral é a preocupação com problemas relacionados às tecnologias proprietárias [106], [122], aos problemas de segurança [42], aos problemas físicos de telecomunicações [62], [155], [186] do que com a especificação correta de programas utilizando a Engenharia de Programas em Sistemas Multimídia e/ou Hipermídia.

Em sistemas de Ensino via rede outros problemas devem ser analisados. Uma grande quantidade de alunos em uma turma acessando as mesmas páginas, ao mesmo tempo, pode congestionar o servidor. Entretanto, é interessante que o acesso do sistema através da rede seja disponibilizado. Neste caso, recomenda-se o acesso restrito através de cadastramento do usuário com senha, caso o curso exija inscrição, e que exista a possibilidade do inscrito possuir um disco compacto contendo as lições e que rode com um navegador de funcionamento local.

Apesar da ampla divulgação de Sistemas de Ensino Virtual [41], [47], [172] [186] ou parcial [64] de aprendizagem via Web [56], [106], [155], [172], são inexistentes as tentativas de resolver os problemas na concepção destes tipos de sistemas.

As melhorias encontradas são devidas mais à aplicação dos paradigmas educacionais [66], [83], [92], [101], [118], [141], [154], [218] do que com os paradigmas construtivos. Mas de nada adianta preparar todo um sistema suportado pelas teorias educacionais se este não funciona corretamente porque não foi bem concebido

em termos computacionais.

Um sistema hipermídia especificado corretamente, em princípio permite a seu usuário ir e voltar para qualquer posição anterior. Isto quer dizer que todos os estados deste sistema devem ser então alcançáveis [165], [166] a partir de outros estados.

5.10 Nós essenciais de um Hipertômato

Um Hipertômato, para ser utilizado na concepção de ambientes de Ensino-aprendizagem via redes, deve conter alguns nós, ditos essenciais. A Figura 5.10 mostra estes nós.

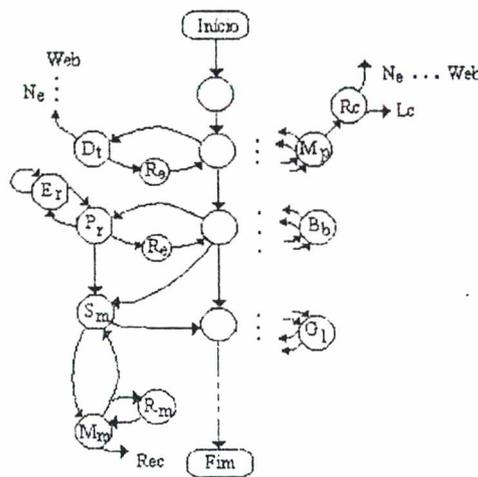


Figura 5.10: Nós essenciais de um Hipertômato

- *nó apresentação*: apresenta o ambiente e oferece informações gerais sobre a autoria, os objetivos, a navegação, os recursos, o endereço;
- *nó mapa* - Mp: oferece um mapa completo do ambiente. Através do nó mapa todos os estados são alcançáveis. Este nó é muito importante para evitar que o aprendiz se confunda ou se perca no ambiente. Funcionalmente, acessar um nó mapa também pode ser comparado ao fato de “folhear um livro rapidamente para ver seu conteúdo do princípio ao fim”;
- *nós unidades*: são apresentadas as unidades com os assuntos escolhidos;
- *nós tópicos e subtópicos*: são extensões dos nós unidades. Os nós subtópicos podem conter exercícios e ligações para os nós de simulação;

- nós de *detalhamento* de unidades, tópicos e subtópicos - Dt: oferecem mais informações que não são relevantes ao assunto principal;
- nó de *bibliografia* - Bb: oferecem a base bibliográfica utilizada nos assuntos abordados;
- nó de *glossário* - Gl: Como em um livro, se o aprendiz quiser saber o significado de alguma palavra considerada incomum ou própria da linguagem abordada, poderá acessar o nó Gl e retornar ao estado anterior;
- nó *local* - Lc: pode conter recursos da rede interna a qual aloca o ambiente. Estes nós devem permitir o acesso a arquivos de dados, listas, acesso ao instrutor, horários de aulas, trabalhos, programas, páginas locais, etc.
- nó *recursos* - Rc: Este nó pode ser ligado a diversos nós tais como arquivos disponíveis (interna ou externamente), nós para listas de discussões, correio eletrônico, painel de horários e informações do instrutor, que deverão ser incorporados de acordo com as necessidades observadas. O nó recursos liga-se ao nó mapa, ao nó recursos locais e a nós externos;
- nó *memória* - Mn: O objetivo do nó memória é oferecer um mapeamento de nós visitados anteriormente. Uma opção de utilização deste nó pode ser a colocação de marcas (“bookmarks”) semelhantes aos livros, na qual o aprendiz pode deixar o sistema e quando voltar verificar onde estava. Este tipo de nó, assim como o nó mapa, é interessante como guia de navegação;
- nó *externo* - Ne: permite a obtenção de recursos disponíveis na Web, com ligações para mecanismos de busca, bibliotecas virtuais, repositórios de artigos, programas acadêmicos e de domínio público (conectores, utilitários, etc.), simuladores, endereços de empresas de programas comerciais e tópicos relacionados;
- nó *prática* - Pr: para práticas e teste de conhecimentos através de exercícios e simulações;
- nó *erro* e ou *realimentação* - Er: para verificações de respostas;
- nó *retorno simples* - Re: para retornar ao nó anterior. Não possui memória. São geralmente feitos com ligações do botão de “voltar” ou com o próprio navegador;
- nó *retorno com memória* - Rm: para armazenar os caminhos percorridos e as ações executadas. Este tipo de nó oferece uma vantagem em relação ao nó de retorno simples pelo fato de poder agir como um “marcador de passos”;

- nó *simulações* - Sm: para desenvolvimento de simulações.

5.11 Metodologia para Construção de Hipertômatos

A seguir é dada a metodologia aconselhada para a construção de Hipertômato . Ao se fazer um Hipertômato é necessário identificar os objetos e definir os morfismos.

1. Definir o domínio de aplicação;
 - Definir os objetos (estados) do Hipertômato;
 - Definir os morfismos (transições);
2. Construir o grafo com nós essenciais;
3. Verificar a existência de não monomorfismos;
4. Verificar se todos os estados são alcançáveis e observáveis;
5. Corrigir a existência de não monomorfismos;
6. Implementar usando uma ferramenta de autoria;
7. Testar e avaliar.

Por exemplo, a página do Projeto ICEM , (Inteligência Computacional no Ensino com Multimídia - <http://www.inf.ufsc.br/~mafa/icem.html>) do Laboratório de Conexionismo e Ciências Cognitivas (L3C), da UFSC foi desenvolvida segundo esta metodologia.

Conforme mostra a Figura 5.11, foi definido o nó inicial como sendo a página “Finalidade” na qual estão contidas as informações sobre o objetivo do projeto.

Em seguida foram definidos os seguintes nós:

- “Histórico”: neste nó são mostrados o histórico do projeto, suas origens e publicações iniciais.
- “Equipe”: este nó contém os nomes dos pesquisadores que desenvolvem trabalhos relacionados ao projeto.
- “Temas”: neste nó são apresentados os temas desenvolvidos pelos membros da equipe.

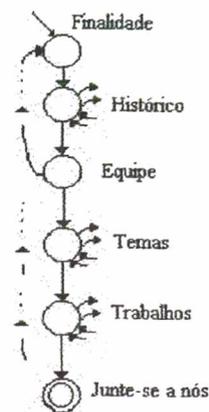


Figura 5.11: Nós essenciais do portal do Projeto ICEM

- “Trabalhos”: este nó contém o repositório dos trabalhos publicados pela equipe.
- “Junte-se a nós”: este nó oferece informações sobre a participação no projeto.

Os nós de detalhamento foram omitidos e são representados pelas linhas pontilhadas. Estes nós se referem a mais informações que o visitante do portal pode encontrar em cada nó principal. A Figura 5.12 mostra a tela da página inicial do projeto.

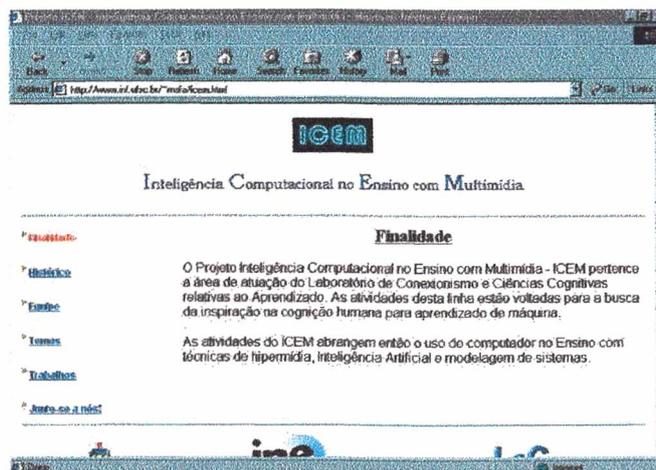


Figura 5.12: Página inicial do portal do Projeto ICEM

5.12 Hipertexto x Ensino de RNA: a inspiração

Esta seção mostra como foram desenvolvidas as implementações de um ambiente de apoio ao Ensino de conceitos introdutórios de Redes Neurais Artificiais. Com a composição do conteúdo inspirada em notas de aulas e trabalhos científicos de professores e alunos do PGEEL e CPGCC da UFSC, foi desenvolvido um tutorial utilizado como apoio acadêmico durante 4 anos (1995-1999).

Na concepção deste sistema organizou-se uma rede de relações entre as informações e criou-se os documentos que constituíram o conteúdo do hipertexto. Esta técnica de apresentação da informação tem um interesse pedagógico importante na formação contextual e na situação de trabalho, segundo Pagano [169]. Todavia, a superposição de informação de forma não linear e os atrativos visuais podem causar impactos mais importantes sobre o usuário em detrimento do próprio conteúdo. Outra consideração muito comum encontrada na Literatura está relacionada a perda do usuário no espaço de informações [178], [179]. A Figura 5.13 mostra uma tela do primeiro programa de Ensino de RNA.

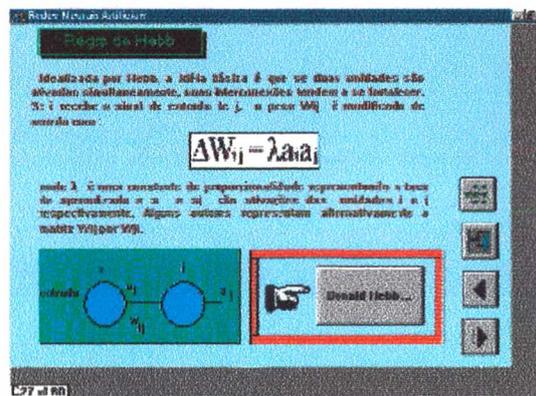


Figura 5.13: Tela de um hipertexto simples

Este primeiro programa, para Ensino de RNA, era atualizado de acordo com as críticas e sugestões solicitadas a seus usuários, que se dispuseram a avaliar e colaborar para sua melhoria. O Ciclo de Vida desse programa terminou quando para atender às sugestões dos usuários, Almeida [5] desenvolveu um segundo protótipo para validar o objeto de estudo sobre modelos teóricos de hipertexto. A formalização do segundo protótipo foi baseada no modelo teórico de hipertexto como autômato apresentado no Capítulo 5.

A Figura 5.14 mostra o grafo de implementação do sistema. As linhas pontilhadas mostram os arcos que ligam-se a outros nós (mostrados na ponta das setas). Os

nós que possuem a mesma configuração e são funcionalmente iguais também foram omitidos. Por exemplo, os nós 2.2 e 4 foram omitidos porque são funcionalmente iguais ao nó 2.

O nó 1 apresenta o ambiente e oferece informações gerais. Está ligado ao nó 2, que possui nó de detalhamento e retorno, ao nó de recursos locais Lc e ao nó de recursos externos Ne. Os recursos se referem a repositórios de programas de simulação de RNA, ligações com outros portais com assuntos referentes às RNA.

O estado inicial $x_0 = 1$. A entrada u_1 = “acionar o nó 1”, conduz como saída y_1 - que é a apresentação de uma página introdutória sobre o ambiente. As transições de estado vão ocorrendo de acordo com o manuseio do ambiente pelo aprendiz.

As unidades com os assuntos diversos sobre RNA foram constituídas pelos nós 1 a 8. Estes assuntos abordam as origens das RNA, sua inspiração biológica, as características das RNA, os tipos de aprendizado, os algoritmos de aprendizado, as topologias das RNA e as suas aplicações.

Os nós (2.1 a 7.8) representam os tópicos e subtópicos. Versam sobre diversos assuntos relacionados aos nós de unidades.

Os nós de A a P oferecem um detalhamento destes tópicos. Um exemplo de nó de detalhamento são informações sobre a linha do tempo das RNA. O detalhamento do nó 3 com a ligação para nós externos do portal é feito pelo nó Dt.

O nó Bb oferece a base bibliográfica utilizada nos assuntos abordados. Nota-se no grafo que estes nós estão ligados a todos os nós unidades 2 a 8. Assim, a base bibliográfica é cruzada com as informações textuais.

No nó Sm o aprendiz desenvolve simulações após atingir o nó Pr (práticas, exercícios) ou ir diretamente ao nó 6.1, que aborda os principais algoritmos de aprendizado das RNA. Mais detalhes sobre este sistema podem ser obtidos em Almeida [16], [15].

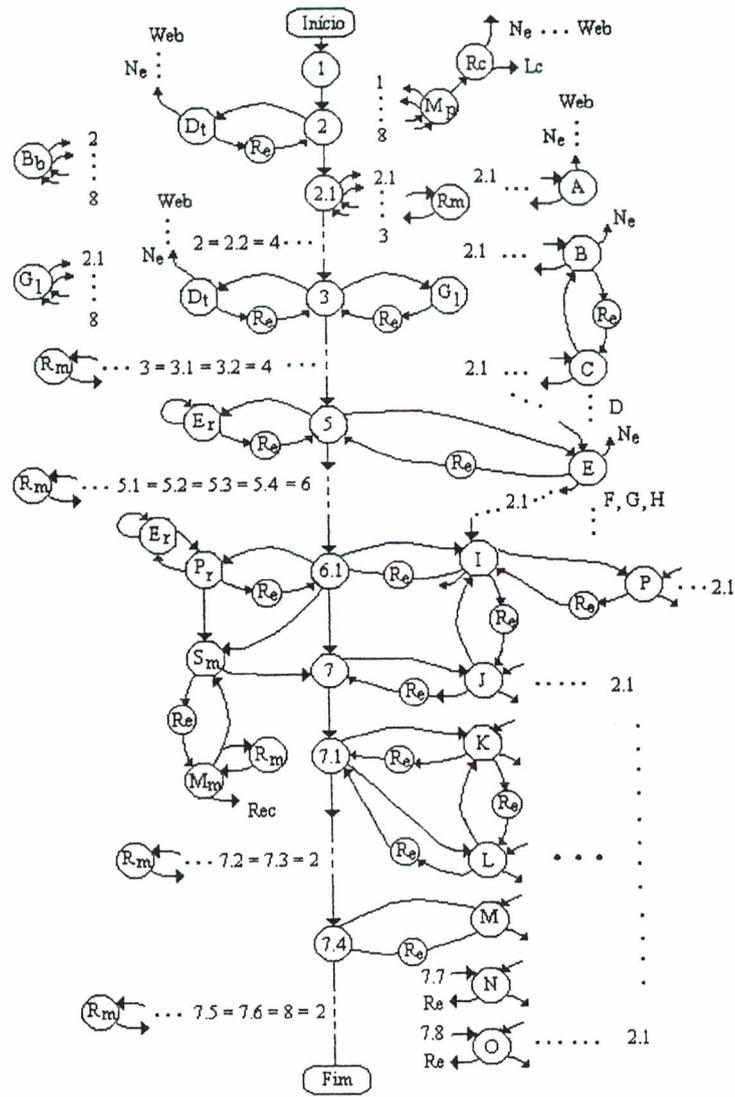


Figura 5.14: Grafo de um Hipertômato para Ensino de Redes Neurais

5.13 Exemplo 1: Um ambiente de Ensino de Computação

Conforme foi dito no Capítulo 1, a autora realizou dois estágios nos quais foi desenvolvido um ambiente de Ensino de Computação para apoio a estas disciplinas. Ao final do curso foi oferecido aos alunos um questionário para coleta de opiniões (*opinário*) para avaliação do ambiente, das preferências dos alunos e do processo de Ensino-aprendizagem. No opinário foi feita a Análise de Oportunidades [82], ou seja, se o cliente (estudante) gosta do Ensino com computador. Para se detectar a personalidade do cliente, deve ser observado o seu o âmago, suas preferências. Assim, foram feitas várias perguntas, algumas contraditórias, propositalmente, para que um primeiro estudo pudesse ser feito.

Na sua primeira versão o ambiente foi desenvolvido utilizando a linguagem HTML. Devido a limitação de recursos da Rede Intranet do Laboratório utilizado (problemas de conectividade, máquinas e programas obsoletos, etc.), a implementação contou com os requisitos mínimos desejáveis para um ambiente deste tipo. Entretanto, na sua especificação (simples por questões de ordem prática), todos os detalhes para se evitar estados inalcançáveis foram considerados. Os alunos acessavam as notas de aulas de “Introdução à Computação” cujo conteúdo foi dividido da seguinte forma:

- Plano e informações sobre a disciplina;
- Histórico sobre computação;
- Conceitos básicos sobre unidades funcionais do computador, algoritmos e programas, interpretadores, compiladores e tradutores, etapas de processamento de programas, entre outros;
- Principais periféricos;
- Sistemas operacionais;
- Paradigmas de programação;
- Redes de computadores;
- Serviços da Internet;
- Tratamento de textos;
- Planilhas;

- Programas de apresentação;
- Sistemas de Base de Dados;
- Sugestões de exercícios;

A Figura 5.15 mostra a tela principal do ambiente modelado cuja concepção foi formalizada utilizando o conceito de Hipermissão como Autômato.

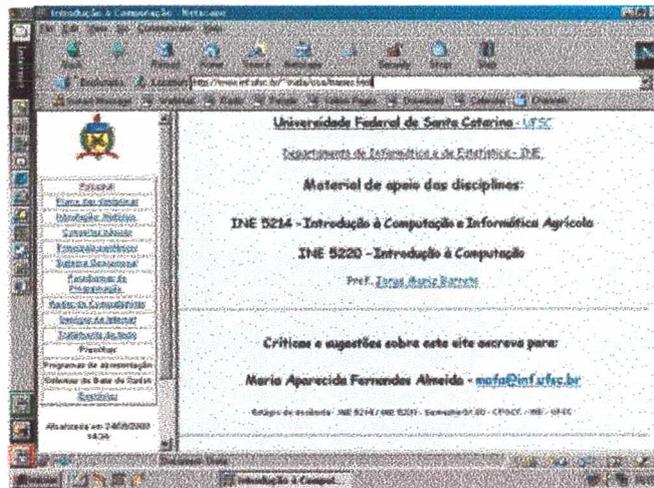


Figura 5.15: Tela de apresentação do ambiente

Esta disciplina era composta por alunos que não pertenciam à área de Computação. Assim, os conceitos foram abordados de maneira informativa, direcionada mais às aplicações do computador do que ao desenvolvimento de sistemas. O curso visava mostrar conceitos de maneira simples mas precisa, para que os alunos pudessem utilizar o computador como uma ferramenta de auxílio na realização de tarefas nas suas áreas de conhecimentos independentemente da tecnologia.

Evitou-se que os conteúdos estivessem também atados a tecnologias proprietárias, mostrando alternativas de programas gratuitos que podem ser obtidos para o uso comum.

A Figura 5.16 mostra uma tela com um tópico de uma aula expositiva sobre *Paradigmas de Programação* que foi ministrada pela Professora na aula teórica. A maioria das Notas de Aulas foram disponibilizadas, não estando o conteúdo preso a um livro-texto.

Durante o período que em ministrava as aulas e usava o material disponível como guia das aulas práticas, a autora pôde perceber as dificuldades e as preferências dos

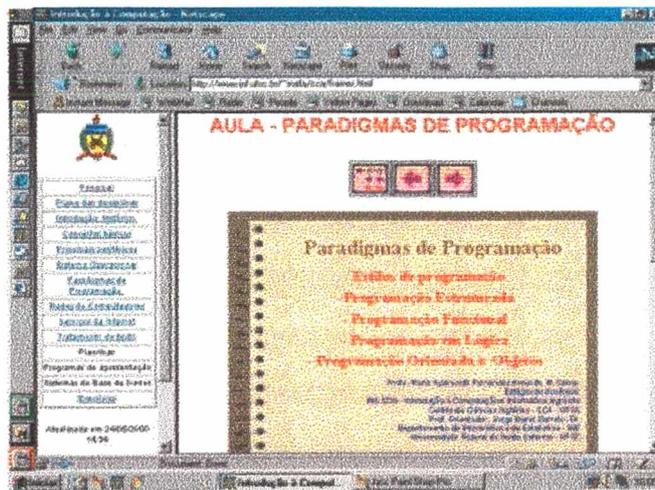


Figura 5.16: Tópico de uma aula expositiva disponível para acesso

alunos, através dos comentários que estes faziam quando acessavam o material na sala de aula. O acesso instantâneo ao conteúdo contribuía no desenvolvimento das aulas práticas. Os alunos, gradativamente, sanavam suas dúvidas abrindo, quando necessário, o tópico relacionado ao desenvolvimento de alguma atividade.

A Figura 5.17 ilustra um exemplo do acompanhamento de uma tarefa dada pela professora (uso do correio eletrônico). Neste caso, os alunos usaram o Telnet² para acessarem suas contas no servidor do CCA e enviarem mensagens utilizando o PINE³.

O sistema foi implementado segundo a modelagem Hipertômato já desenvolvida por Almeida [5] para Ensino de IA. Uma primeira preocupação foi a concepção correta do sistema. Um sistema de Ensino, hipertexto (ou hipermídia), no qual todos os estados são alcançáveis e observáveis aumenta a eficiência do processo de Ensino aprendizagem via computador, principalmente quando se usa sistemas na rede mundial. A concepção do sistema é simples pois o sistema é limitado pelas máquinas e pela rede acessadas pelos alunos.

A Figura 5.18 mostra o grafo de implementação do sistema.

Este sistema foi testado com a colaboração dos alunos, conforme será mostrado na seção 6.3 do Capítulo 6.

²Programa de emulação remota para Windows.

³Programa para correio eletrônico do UNIX

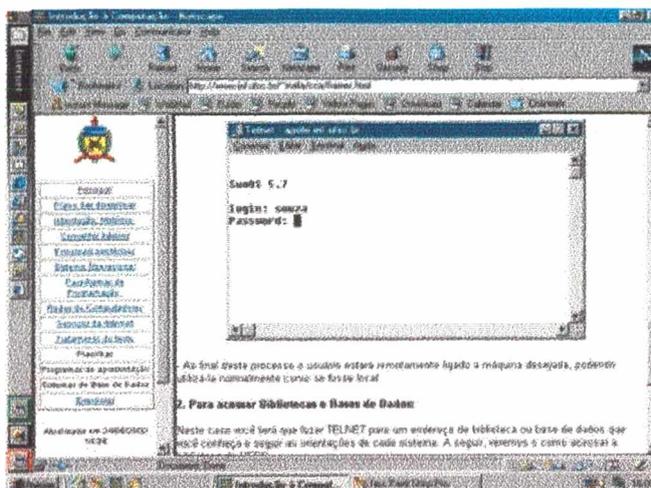


Figura 5.17: Acompanhamento de uma tarefa dada

5.14 Exemplo 2: Hipertômato para Ensino de Neurofisiologia

O modelo Hipertômato foi utilizado por Vedana [214] na concepção de um protótipo para Ensino de conceitos introdutórios de Neurofisiologia na Universidade do Planalto Catarinense. A Figura 5.19 mostra o grafo de implementação desenvolvido por Vedana[214], utilizando a modelagem Hipertômato.

Neste sistema, o usuário inicia a navegação do sistema tipo "espinha dorsal" no nó 1 (estado inicial) no qual é feita a sua apresentação com recursos na rede (internos e externos). Em seguida, o usuário poderá seguir o percurso nó 2 que contém os tópicos de apresentação da célula biológica. Deste, poderá percorrer o nó 3 que trata o neurônio e em seguida o nó quatro (sistema nervoso). Nota-se que os nós possuem retorno, e conexões para outros nós (nós de detalhamento). A partir do nó 4, o usuário poderá percorrer o nó 5 que apresenta o mapa do portal. Finalmente, o usuário atingirá o nó 6 (estado final) e pode sair do sistema.

Após a modelagem seguiu-se a implementação com a ferramenta de autoria. Inicialmente, foi definido o conteúdo do ambiente que consiste em um bloco de apresentação no qual pode ser encontrada uma moldura e um mapa de navegação. Isto permite ao usuário selecionar o assunto de seu interesse. Um segundo bloco apresenta as informações sobre a célula biológica, sua composição e suas principais partes. O sistema nervoso é abordado no terceiro bloco com informações sobre os avanços no campo da Neurologia e sua organização geral. O quarto e último

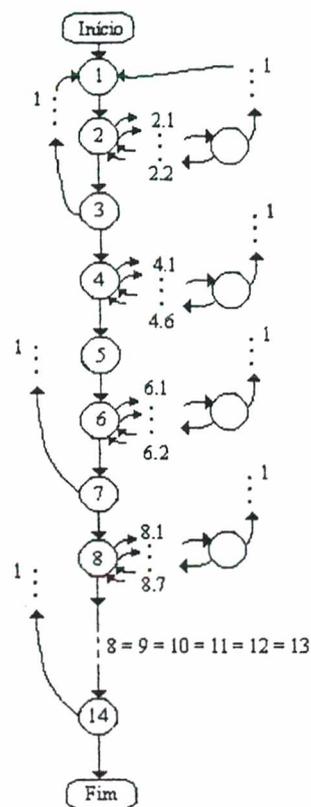


Figura 5.18: Grafo de um Hipertômato para Ensino de Computação

bloco contém as informações sobre o mecanismo de potencial de ação em uma célula nervosa e como as informações são transmitidas entre os neurônios. Estes blocos podem conter um ou mais nós de informações conectados a outros documentos do mesmo bloco ou a blocos distintos.

O sistema consiste de quatro blocos funcionais:

- Apresentação do conteúdo;
- A célula biológica;
- Sistema nervoso;
- Potencial de ação.

Uma vantagem da modelagem Hipertômato é a flexibilidade no entendimento e nas mudanças no grafo, o que facilita atualizações. O modelo Hipertômato permite a visualização de todos os nós (estados) do sistemas e das ligações (transições de

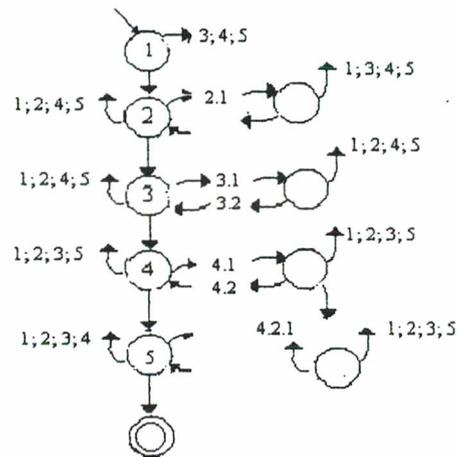


Figura 5.19: Grafo de um protótipo para Ensino de Neurofisiologia [214]

estado) que ocorrerão quando o usuário navegar no sistema. A grande vantagem da concepção do sistema através desta modelagem é a garantia de que todos os estados do ambiente sejam alcançáveis, ou seja, muitos dos problemas de navegação (perda do usuário no espaço de informações, páginas não encontradas), que, na maioria das vezes, dependem dos aspectos construtivos dos sistemas podem ser evitados. O trabalho de Vedana [214] apresenta os resultados obtidos na avaliação deste sistema por alunos da UNIPLAC.

Pode-se dizer, que a modelagem Hipertômato permite identificar, a priori, em termos construtivos, todos os problemas cruciais que possam ocorrer na navegação em sistemas hipertexto e/ou hipermídia. A introdução da teoria das categorias visa simplificar ainda mais esta formalização para que as especificações sejam independentes do domínio e da própria tecnologia.

Para tentar contribuir com a resolução destes problemas, que inevitavelmente tomará grandes proporções devido às constantes mudanças tecnológicas [74], [90], [159], [210], é que se propõe nesta tese a aplicação de métodos, que por serem formais poderão ser utilizados por qualquer Cientista da Computação.

Capítulo 6

Hipertômato e Ciclo de Vida

*“The only way to rectify our reasonings
is to make them as tangible as those of
the mathematicians, so that we can find
our error at a glance, and when there are
disputes among persons we can simply say,
‘Let us calculate ... to see who is right.’ ”*

Leibniz, The Art of Discovery.

Este Capítulo relaciona o Ciclo de Vida de um Programa Educacional com a modelagem Hipertômato apresentando, em termos práticos, uma pesquisa efetuada com diversos alunos que utilizaram o sistema descrito na seção 5.13 do Capítulo 5. São apresentados também os resultados de uma pesquisa geral, efetuada com a colaboração de alunos de diversos cursos na UFSC, para a observação do perfil de uma potencial comunidade de usuários de sistemas computadorizados para apoio ao Ensino.

6.1 Ciclo de Vida em Programas Educacionais

O conceito de Ciclo de Vida em programas educacionais é um conceito novo, nascido, de um modo geral, na Engenharia de Programas, para definir as fases de desenvolvimento de programa. Na Literatura são sugeridas diversas abordagens [135] sobre Ciclo de Vida de programas para uso geral. Sugere-se aqui, que o Ciclo de Vida de um programa educacional seja dividido em:

- Análise de Oportunidades : são estudadas as características relevantes do programa relacionadas com a finalidade deste e do interesse de uma comunidade de usuários;
- Análise Funcional : após saber se o sistema tem uma potencial comunidade de usuários torna-se necessário saber quais as funcionalidades desejadas para este programa;
- Análise Orgânica: divide-se o programa em módulos;
- Implementação: tem-se o desenvolvimento construtivo do sistema utilizando alguma linguagem e/ou ferramenta de programação;
- Teste: consiste na verificação de erros e avaliação;
- Manutenção: implica na atualização do programa, melhoria da interface, desempenho, etc.;
- Morte: finaliza o Ciclo de Vida do programa.

A realidade brasileira mostra que “o uso do computador no Ensino ainda está engatinhando”... portanto, muito ainda “pode” (e tem que) ser feito para sua melhoria. É necessário observar as diferenças regionais, culturais, econômicas e democratizar o Ensino com computador de acordo com as condições institucionais e pessoais. Os professores e estudantes brasileiros necessitam de programas educacionais brasileiros e não de adaptações de programas estrangeiros.

Uma primeira pesquisa visou a detecção de erros, manutenção e verificação da aceitabilidade de um protótipo desenvolvido para Ensino de Computação nos Cursos de Agronomia e Engenharia de Aqüicultura do CCA na UFSC. Ao final da disciplina, os alunos responderam um questionário de opiniões, o que motivou a melhoria do opinário para ser aplicado no início (*opinário 1*) e final (*opinário 2*) da disciplina de ICC em vários cursos da UFSC. Apesar do caráter descritivo, a intenção da pesquisa não foi desenvolver nenhum estudo estatístico mas apenas observar algumas características e preferências de um público alvo para o uso de programas educacionais hipermídia mostrando que é oportuno o desenvolvimento de programas educacionais para o apoio ao Ensino de disciplinas de ICC para os diversos cursos. A modelagem Hipertômato oferece o ferramental necessário para o desenvolvimento destes sistemas.

6.2 Resultados de uma pesquisa geral

No ano de 2001 foi feita uma pesquisa com 370 alunos que cursaram a disciplina “Introdução à Ciência da Computação” (ICC), em várias turmas de diversos cursos de graduação na Universidade Federal de Santa Catarina (UFSC). A disciplina ICC é comum a muitas áreas e foi constatado que as turmas eram bem heterogêneas. Entretanto, identificou-se que a maioria dos alunos eram oriundos dos cursos de Engenharia Sanitária e Ambiental, Engenharia de Produção Civil, Engenharia Mecânica, Engenharia de Aqüicultura, Agronomia e Bacharelado em Física.

Os opinários foram passados aos alunos no início (Opinário 1) e no final (Opinário 2) dos semestres letivos. Os resultados obtidos são mostrados nas Tabelas 6.1 e 6.2 em termos de percentuais. Para uma melhor compreensão as perguntas são transcritas aqui:

- *Você já utilizou o computador anteriormente ?*
 - Sim*
 - Não*

- *Se sim, há quanto tempo utiliza o computador?*
 - Menos de 6 meses*
 - 1 ano*
 - Mais de 2 anos*

- *Atualmente, quantos dias num mês você utiliza o computador?*
 - Menos de 5 dias*
 - De 10 a 15 dias*
 - Mais de 15 dias*

- *Quantas horas costuma utilizar o computador diariamente?*
 - Menos de 1 hora*
 - 2 a 4 horas*
 - Mais de 4 horas*

A pesquisa mostrou que a maioria dos alunos eram usuários de computadores antes de ingressarem na disciplina de ICC. O tempo médio de uso era superior a dois anos. Uma exceção foi encontrada no curso de Agronomia, no qual 7 % dos alunos

não tinham utilizado o computador anteriormente. Embora a utilização fosse de mais de quinze dias em um mês, o uso limitava-se a menos de uma hora diária. De um modo geral, após a disciplina, ocorreu um ligeiro aumento nas horas utilizadas em todos os cursos.

Aspectos	Respostas	Opinário 1 (%)	Opinário 2 (%)	Média (%)
Utilização Anterior	Sim:	99	99	99
	Não:	1	1	1
Tempo utilização	Menos 6 meses:	4	4	4
	1 ano:	11	11	11
	Mais de 2 anos	85	85	85
Utilização Mensal	Menos 5 dias:	14	17	16
	10 a 15 dias:	28	33	30
	Mais 15 dias:	58	50	54
Utilização Diária	Menos de 1 hora:	52	48	50
	2 a 4 horas:	38	41	40
	Mais de 4 horas:	10	11	10
Desconhece o termo	V. Neumann:	95	93	94
	Laço:	93	88	90
Utilização Assistentes	Com freqüência:	13	18	16
	Às vezes:	43	25	34
	Raramente:	44	57	50
Ensino melhor	Com computador:	87	91	88
	Sem computador:	13	9	12
Livro ou Hipertexto?	Livro:	67	68	68
	Hipertexto:	33	32	32
Exercícios	Impresso:	38	40	38
	"On-line":	62	60	62
Ensino menos fatigante ?	Sim:	45	40	42
	Não:	55	60	58
Ler na tela ou imprimir?	Ler:	37	39	38
	Imprimir:	63	61	62
Favorável ao material local	Sim:	84	91	87
	Não:	16	9	13
Críticas e Sugestões	Opinaram:	8	18	13
	Não opinaram:	92	82	87

Tabela 6.1: Resultados dos Opinários 1 e 2

Nas Engenharias (Sanitária, Mecânica e de Produção) e no curso de Física, mais da metade dos alunos utilizavam o computador acima de 15 dias no mês. Nos cursos da área de Ciências Agrárias (Agronomia e Aqüicultura) este uso mensal não era tão intenso assim. Nota-se que na utilização diária do computador a tendência geral é o uso de menos de 1 hora por dia.

Na Engenharia Sanitária o estudo demonstrou que o número de dias de uso

mensal do computador aumentou no final do período e que mais da metade dos alunos utilizavam o computador no máximo 1 hora por dia. A taxa de utilização de 2 a 4 horas diárias aumentou no final da disciplina. Este aumento pode ser explicado porque muitos alunos consideraram o uso do computador no laboratório. Neste ponto foi detectado uma falha no opinário que não especificou onde o aluno manipulava o computador diariamente.

Na Agronomia a maioria dos alunos utilizavam o computador há mais de 2 anos. Quase metade dos alunos utilizavam o computador mais de 15 dias no mês. Entretanto, a porcentagem de utilizadores de menos de 5 dias no mês aumentou. Também a utilização do computador em menos de 1 hora por dia aumentou cerca de 10 % no final da disciplina.

Na Engenharia de Produção, é interessante notar que os alunos cuja porcentagem de utilização do computador era menor do que 1 hora diária (44,44 %) diminuiu para 27,78 %. Um aumento ocorre quando para o uso de 2 a 4 horas no mês. Este aumento também foi verificado na utilização por mais de 4 horas. Nota-se que os alunos utilizam mais o computador e com um pouco mais de tempo.

- *Marque com um X os principais usos que faz do computador?*

Digitar textos.

Desenhar figuras ou manipular fotos.

Jogar.

Escutar música.

Assistir filmes e clipes.

Navegar na Internet (pesquisas, entretenimento, informações gerais).

Ler e enviar mensagens eletrônicas.

Usar programas de comunicação na Internet ('bate-papos', conferências).

Efetuar cálculos (planilhas, simulações, etc.).

Aprender utilizar programas (educativos, de uso geral, etc.)

Construir seus próprios programas.

Outros.

Aspectos	Respostas	Eng. Sanit.	Agron.	Eng. Aquic.	Eng. Mec.	Eng. Prod.	Fis.
Utilização Anterior	Sim:	100	93	100	100	100	100
	Não:	0	7	0	0	0	0
Tempo utilização	Menos 6 meses:	0	3	12	4	0	0
	1 ano:	6	10	16	12	8	16
	Mais de 2 anos	94	87	72	84	92	84
Utilização Mensal	Menos 5 dias:	10	24	35	6	3	7
	10 a 15 dias:	33	34	39	33	22	18
	Mais 15 dias:	57	42	26	61	75	75
Utilização Diária	Menos de 1 hora:	57	60	65	55	36	29
	2 a 4 horas:	34	36	27	37	50	47
	Mais de 4 horas:	9	4	8	8	14	24
Principais Usos	Digitação:	15	19	16	12	14	15
	Navegação:	19	18	15	18	20	19
	E-mail:	18	17	15	16	18	17
	Entretenimento	11	10	11	12	8	9
Desconhece o termo	V. Neumann:	94	77	87	89	89	92
	Laço:	83	88	85	77	67	70
Utilização Assistentes	Com freqüência:	20	19	15	16	19	15
	Às vezes:	44	24	30	35	30	28
	Raramente:	36	57	55	49	51	57
Ensino melhor	Com computador:	94	87	90	96	83	73
	Sem computador:	6	13	10	4	17	27
Livro ou Hipertexto?	Hipertexto:	29	38	45	26	31	29
	Livro:	71	62	55	74	69	71
Exercícios	Impresso:	21	43	38	30	25	38
	"On-line":	79	57	62	70	75	62
Ensino menos fatigante ?	Sim:	54	42	34	51	31	41
	Não:	46	58	66	49	69	59
Ler na tela ou imprimir?	Ler:	25	33	38	39	44	58
	Imprimir:	75	67	62	61	56	42
Favorável ao material local	Sim:	86	83	93	86	78	82
	Não:	14	17	7	14	22	18
Críticas e Sugestões	Opinaram:	15	16	14	6	0	8
	Não opinaram:	85	84	86	94	100	92

Tabela 6.2: Resultado em cada curso

O gráfico¹ da Figura 6.1 mostra os principais usos do computador que são: navegar na rede, ler e enviar mensagens eletrônicas e digitar textos. As atividades de entretenimento (escutar música e jogar) seguem estas preferências. Em seguida, vem a utilização dos programas de comunicação, o desenho de figuras, o aprendizado na utilização de programas e na efetuação de cálculos. Finalmente, segue o uso de outras atividades, construir seus próprios programas, assistir filmes.

Em todos os cursos, os principais usos do computador incluem navegação na

¹Os gráficos no lado superior de todas as figuras mostradas referem-se ao opinário 1 e os gráficos

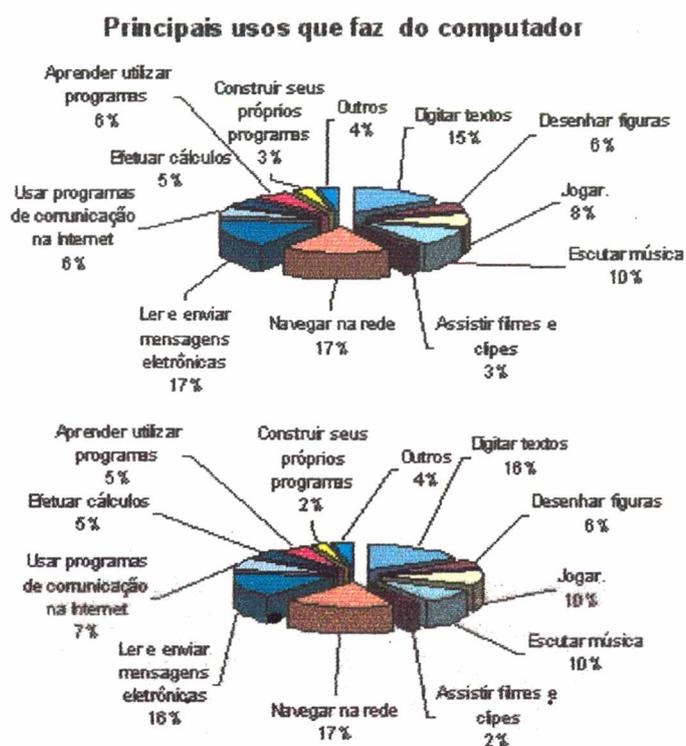


Figura 6.1: Principais usos do computador

rede, leitura e envio de mensagens eletrônicas e digitação de textos. O uso para entretenimento (escutar música e jogar), vem logo em seguida na seqüência de preferências. Entretanto, nos cursos do CCA foi detectado que os principais usos do computador são a digitação de textos, seguida pela navegação na rede, envio de mensagens e entretenimento.

Na Agronomia, as preferências de uso do computador são digitar textos, navegar na rede, enviar e receber mensagens eletrônicas, respectivamente. Seguem então as atividades de entretenimento (escutar música, jogar). O uso de planilhas para efetuar cálculos aumenta significativamente no decorrer do curso. Isto pode estar relacionado com as habilidades desenvolvidas pelos alunos no uso de aplicações para sua área de interesse. Por exemplo, na Física a utilização do computador para efetuar cálculos aumentam em cerca de 5 % ao final da disciplina.

Notou-se que porcentagem na Aqüicultura, embora seja muito baixa (6 %) uma parcela dos alunos usa para construir os próprios programas. Os jogos, entretenimento no lado inferior das respectivas figuras referem-se ao opinário 2.

mentos, seguem a preferência com 5 %. A porcentagem de uso na aprendizagem também aumentou em 15 % no final da disciplina. Neste caso, pode-se dizer que houve um certo interesse no uso de programas educacionais pois estes alunos utilizaram o ambiente descrito na seção 5.13. Embora a Agronomia também tenha utilizado o mesmo ambiente este aumento foi apenas de 5 %.

- *O que você acha que significa na computação o termo 'von Neumann'. Marque com X as opções que julgar mais provável:*

- Um tipo de computador muito moderno.*
- Um componente eletrônico do computador inventado por John von Neumann.*
- O nome de um fabricante de computador.*
- O nome de um inventor de um sistema operacional.*
- Nenhuma das anteriores.*
- Desconhece o termo.*

- *O termo 'laço' num programa refere-se a:*

- A união de um programa a outro em forma de 'laço'.*
- Um tipo de estrutura de controle utilizada na programação de computadores.*
- Uma mensagem de erro que o computador emite quando ocorre um erro, fazendo com que o mesmo volte a seu estado anterior.*
- Desconhece o termo.*

A maioria dos alunos dos cursos são apenas usuários do computador e desconhecem termos avançados na computação. É interesse notar que na Agronomia a porcentagem daqueles que afirmam desconhecer os termos perguntados é mais baixa (77 %) que as outras e isto pode estar relacionado ao fato de que muitos marcaram respostas diversas. Os alunos dos cursos das Engenharias (Mecânica e Produção) e Física demonstraram mais conhecimento do que seja o termo "laço em um programa". Presume-se que estes conceitos foram abordados com mais profundidade nestas turmas. Além disso, nas Engenharias nota-se uma utilização maior do computador na confecção de programas, elaboração de planilhas e utilização de programas educacionais.

A Tabela 6.1 mostra que mais de 90 % dos alunos desconhecem termos avançados na computação tais como "V. Neumann" e "laço" em um programa. Este resultado

era esperado pois, trata-se de uma disciplina introdutória em cursos que não são da área de computação. Alguns tópicos são abordados superficialmente e outros com mais profundidade, dependendo do curso. Há um aumento nas respostas corretas em relação ao que seja “laço” em um programa.

- *Utiliza muito os “assistentes” que auxiliam em tarefas tais como: elaborar um currículo, escrever uma página para Internet, enviar um Fax, conectar a Internet, instalar programas, resolver problemas de configuração de programas no computador.*

() Com freqüência

() Às vezes

() Raramente

O uso de assistentes, destinados ao auxílio na manipulação de programas, não é muito difundido entre os alunos. O uso de assistentes está relacionado aos limites superior e inferior do grau de conhecimento em computação desses alunos. Quanto menos dominam o computador menos usam assistentes. Naturalmente, usuários iniciantes preferem o auxílio de um ser humano. Os usuários avançados tendem a ler livros e manuais técnicos sobre o assunto e/ou programa de seu interesse.

Em todos os cursos, notou-se que a utilização de assistentes não ultrapassa os 20 %. Mais da metade dos alunos raramente utiliza assistentes. Na Engenharia Sanitária a preferência pelo uso de assistentes aumentou em cerca de 10 % no final da disciplina. Na Agronomia, inicialmente, quase 70 % afirmam que raramente utilizam assistentes, mas esta porcentagem diminuiu, o que pode demonstrar que estes passam a utilizar mais assistentes pois as respostas de utilização “às vezes” e “com freqüência” aumentam. Na Aqüicultura, o uso de assistentes com freqüência diminuiu e a opção ‘as vezes’ aumenta e “raramente” mantém-se estável.

Na Engenharia Mecânica, no primeiro opinário, mais da metade diz que raramente utiliza assistentes e no segundo opinário esta porcentagem diminuiu e aumentou o número de alunos que utiliza ‘as vezes’ e com freqüência (12 %).

Na Engenharia de Produção, a porcentagem de alunos que “raramente” utilizam assistentes diminuiu e daqueles que “às vezes” utilizam assistentes aumenta. Na Física também houve uma diminuição no uso de assistentes. A metade afirma utilizar ‘às vezes’.

- *O computador também tem sido usado como coadjuvante no Ensino geral. Você acha que o ensino com computador numa disciplina é melhor do que o ensino sem computador?*

Melhor ensino COM computador

Melhor ensino SEM computador

A Tabela 6.1 mostra que no primeiro opinário, 87 % dos alunos acham que é melhor o Ensino numa disciplina com auxílio do computador. Esta porcentagem aumenta para 91 % no segundo opinário.

Na Engenharia Sanitária, a opinião dos alunos quanto à utilização do computador como coadjuvante no Ensino permaneceu inalterada sendo que a maioria deles são favoráveis ao Ensino com computador. Na Agronomia, a maioria (quase 90 %) afirmou que o melhor é ensino com o computador. Na Aqüicultura, é interessante notar que a maioria prefere o ensino com o computador, embora a porcentagem daqueles que preferiram o ensino sem computador dobre no final da disciplina.

Na Engenharia Mecânica, a opinião pouco muda em relação ao Ensino com computador. Em média 96 % acham que o ensino com computador é melhor. O mesmo não acontece com a Engenharia de Produção. Inicialmente, mais de 80 % se dizem a favor do computador como coadjuvante no ensino. Entretanto, no decorrer da disciplina, 30 % dos entrevistados mudam suas opiniões dizendo que não são mais favoráveis ao Ensino com computador. Uma reflexão pode ser feita: será que as dificuldades encontradas no manuseio do computador durante a disciplina diminuiu as expectativas dos estudantes em relação ao uso do computador no Ensino?

De maneira geral, a maioria dos alunos concorda que o Ensino é melhor com computador mas esta taxa é mais alta na Física. Inicialmente, quase 80 % dizem que melhor ensino com o computador. No final esta a porcentagem sobe para 92 %. Em média, 73 % dos alunos da Física acreditam que o Ensino com o computador é melhor. Este resultado é muito interessante porque os alunos desta turma de Física não tinham aulas de laboratório na disciplina. Será que este resultado não mostra uma expectativa positiva dos alunos em relação ao Ensino com computador ?

- *Supondo-se que você cursasse uma disciplina contendo material didático na Internet. Neste caso, o conteúdo seria apresentado na forma de hipertexto (páginas que se ligam a outras páginas na Internet). Você acha que seria melhor que o conteúdo de uma disciplina fosse nesta forma ou na forma de um livro convencional? (Note que num livro, uma rápida "folheada" permite visualizar a existência de figuras, a extensão do livro, etc. Assim, você tem uma noção da quantidade de material que tem pela frente, o que não acontece com o ensino com computador na forma de hipertexto).*

Livro

Hipertexto

A pesquisa revelou que independentemente do meio de entrega do conteúdo em uma disciplina (até mesmo de Ensino de computação), os alunos preferem o livro clássico, convencional e impresso. Observou-se os seguintes comentários mostrados na seção 6.3:

“... O material é bem escrito e apresentado. Em contrapartida, ainda não me acostumei a deixar de lado o lápis e o papel...”

“... Com certeza é uma forma de auxílio ao estudo obtido dentro da sala de aula mas não seria tão eficiente se fosse apresentado como único recurso para aprendizagem...”

A Figura 6.2 mostra a preferência pelo livro convencional para entrega do conteúdo em uma disciplina. Apenas 33 % dos alunos preferiram o hipertexto. Esta porcentagem diminuiu cerca de 1 % no final da disciplina.



Figura 6.2: Livro ou hipertexto?

Em geral, em todos os cursos, os alunos preferem o livro convencional ao hipertexto. Na Engenharia Sanitária, cerca de 70 % dos alunos demonstram claramente que preferem o livro convencional. A inclusive a porcentagem daqueles que inicialmente opinaram sobre a preferência pelo hipertexto decresceu no decorrer da disciplina. Todavia, na Agronomia, o número de alunos que preferem o livro convencional diminuiu no final da disciplina. Vale ressaltar que as mudanças de opinião não foram muito significativas: cerca de 10 % dos alunos passaram a achar o hipertexto melhor do que o livro convencional. O mesmo aumento ocorreu na Aqüicultura pois a preferência pelo hipertexto aumentou também em 10 % no final da disciplina. Uma explicação para este aumento pode estar no fato de que estes alunos receberam efetivamente o apoio didático do ambiente descrito na seção 5.13. Isto parece ser um bom resultado da concepção e utilização do ambiente em questão. Na Engenharia Mecânica, inicialmente, 44 % disseram preferir o livro convencional. Esta preferência praticamente dobrou ao final da disciplina. Estes alunos utilizaram computadores nas mais diversas atividades, mas rejeitaram a idéia do hipertexto no Ensino. Na

Engenharia de Produção e na Física a preferência pelo livro convencional aumentou em cerca de 10 %.

- *Supondo-se que você cursasse uma disciplina contendo material didático na Internet. O que você acha que seria mais interessante, fazer exercícios de múltipla escolha em material impresso ou utilizar páginas na Internet contendo exercícios (com opções de múltiplas escolhas). Imagine que nas páginas da Internet, ao final de cada exercício você saberia se escolheu a resposta certa ou não.*

() *Exercícios - material impresso*

() *Exercícios - material "on-line" via Internet*

No opinário 1, cerca de 62 % dos alunos responderam que seria interessante fazer exercícios "on-line". No opinário 2, esta porcentagem diminui em 2 %. Isso mostra que o computador pode ser interessante como coadjuvante no processo de Ensino, principalmente no que se refere a práticas, exercícios e simulações. Em geral, os alunos preferem exercícios "on-line" a exercícios impressos. Na Engenharia Sanitária, cerca de 79 % dos alunos demonstraram a preferência por exercícios "on-line" mas a porcentagem decresceu cerca de 1 % no segundo opinário.

Na Agronomia, inicialmente as opiniões se dividiam igualmente entre exercícios impressos e "on-line". Ao final da disciplina tem-se um aumento de 10 % nas preferências por exercícios "on-line".

Na Engenharia de Aqüicultura, a preferência pelo exercícios "on-line" decresce quase 30 %. Este grande decréscimo pode estar relacionado a problemas de conectividade da rede do CCA.

Na Engenharia de Produção, a maioria prefere exercícios 'on-line' na rede. Essa porcentagem aumenta no final da disciplina. Porém, na Engenharia Mecânica, as preferências pouco mudam ao final da disciplina. Na Física, as preferências não mudam muito e, em ambos os casos, 60 % dos alunos preferem exercícios com material 'on-line'.

Grande parte desses alunos não acham que o processo de aprendizagem com o auxílio do computador seja menos fatigante. No primeiro opinário 55 % dos alunos não acharam que o processo de aprendizagem com computador é menos fatigante. O interessante é que no final da disciplina a porcentagem destes alunos aumenta em 5 %.

- *Você acha que a aprendizagem por computador é mais conveniente pois quando se cansa de manipular o computador desliga-o e só retorna quando tem vontade. Acha que é então que este processo é menos fatigante?*

Acha que a aprendizagem por computador é menos fatigante?

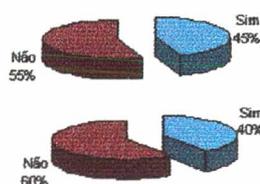


Figura 6.3: Ensino menos fatigante?

() *Sim*

() *Não*

A metade dos alunos dos cursos de Engenharia (Sanitária e Mecânica) acredita que a aprendizagem com o computador é menos fatigante. Os alunos dos outros cursos discordam. Entretanto, na Engenharia Sanitária a porcentagem de alunos que acreditam que a aprendizagem com o computador é menos fatigante diminuiu ao final da disciplina. O mesmo ocorre com a Engenharia Mecânica.

- *Para aprender você prefere ler o material diretamente na tela do computador ou prefere imprimir o que está na tela no papel? Note que a utilização do computador você tem as facilidades das ligações do hipertexto.*

A) *Ler diretamente na tela*

B) *Imprimir o material para leitura*

Conforme mostrado na Figura 6.4 no opinário 1, 63 % dos alunos preferem imprimir o material do que ler diretamente na tela. A porcentagem de alunos que preferem imprimir sofre um pequeno decréscimo, no opinário 2, em cerca de 2 %. Ocorreu uma pequena motivação para leitura na tela? No caso do curso de Física, a leitura na tela atinge 58 %.

Inicialmente, as opiniões na Física, se dividiam igualmente entre leitura na tela e impressão. No final da disciplina, a preferência pelo livro convencional aumentou em 11 %. O mesmo ocorreu com a Engenharia de Aquicultura. Entretanto, o aumento da preferência pela impressão na tela aumentou mais (cerca de 20 %).

Na Engenharia Sanitária a preferência pela leitura na tela diminuiu ao final da disciplina. O contrário ocorre na Agronomia pois ao final da disciplina, a porcentagem de alunos que preferem ler na tela aumenta em cerca de 5 %.

Na Engenharia Mecânica, a porcentagem de alunos que preferem a impressão diminuiu em 10 % ao final da disciplina. O mesmo ocorre com a Engenharia de

Prefere ler o material diretamente na tela do computador ou prefere imprimir o que está na tela no papel?

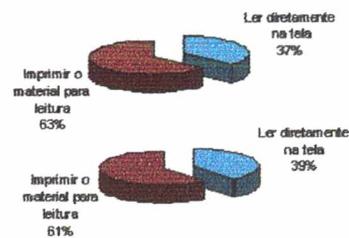


Figura 6.4: Ler na tela ou imprimir?

Produção, cuja preferência pela impressão diminui. Entretanto, na Engenharia de Produção ocorre um aumento significativo na preferência pela leitura.

- *O acesso de um material didático de uma disciplina, via rede (no laboratório ou através de uma conexão de acesso discado via modem conectado na linha telefônica), muitas vezes pode ser lento. A instalação das páginas localmente, no disco rígido do computador, eliminaria os problemas de lentidão e conectividade. Você acha que seria interessante ter um CD-ROM contendo o material, instalar num computador na sua casa ou escritório, para manipulá-lo a qualquer hora, independente das conexões da rede do laboratório ou de acesso discado?*

Sim

Não

No primeiro opinário, 84 % dos alunos acharam que seria interessante que no início da disciplina fosse distribuído um disco compacto contendo o material em hipertexto. No final da disciplina esta porcentagem aumenta para 91 %. Esta é a mesma porcentagem de alunos que preferem ensino com computador.

Na Engenharia Sanitária, inicialmente, mais de 90 % dos alunos são favoráveis ao uso do disco compacto com o conteúdo da disciplina, entretanto esta preferência diminui cerca de 10 % no final da disciplina.

Na Aqüicultura, a maioria (mais de 93 %) é favorável ao disco compacto e esta preferência fica inalterada no final da disciplina.

Na Engenharia Mecânica, mais de 80 % dos alunos são favoráveis ao uso do disco compacto e a quantidade dos desfavoráveis diminui 12 % no final da disciplina. Na Engenharia de Produção, 80 % acha interessante a entrega do conteúdo no disco

compacto. Entretanto, ao contrário dos outros cursos, esta porcentagem de alunos diminui no final da disciplina.

Na Física cerca de 95 % eram favoráveis ao disco compacto. Esta preferência aumentou exclusivamente para 100 % no final da disciplina.

Vale ressaltar que, embora sejam favoráveis ao material impresso, os alunos não demonstram ser contra o Ensino com computador. A pesquisa parece indicar que o computador pode ser muito útil como apoio didático. A obtenção de um disco compacto com o conteúdo facilitaria este processo.

- *Você acha que esquecemos de perguntar alguma coisa interessante? Coloque-a aqui, por favor..*

A) *Opiniões*

B) *Omissos*

No opinário 1, quando perguntados sobre perguntas que eles gostariam que tivessem sido feitas apenas 8 % opinaram. No opinário 2 esta porcentagem aumenta em 10 %. Algumas destas opiniões foram transcritas aqui:

- *Você tem computador?*
- *Você gosta de manipular ou não um computador?*
- *A Internet é uma opção única de pesquisa para você?*
- *A maioria ainda não tem o computador que dificulta o estudo em hipertexto após o horário de aula. Contudo, sanada esta dificuldade o hipertexto é a melhor forma de se estudar, preferencialmente em CD-ROM.*
- *Você acha necessário monitores na sala de Informática? Onde eles estão? Como encontrá-los?*
- *Se tendo a disposição de ler um livro na Internet, você deixaria de comprá-lo?*
- *Se fez algum tipo de curso, há quanto tempo?*
- *Se o aluno possui computador em casa com Internet?*
- *Na verdade, tenho poucos conhecimentos de Informática para poder sugerir ou criticar alguma coisa.*
- *O questionário deveria também ser aplicado no final da disciplina.*

- *Acessibilidade da Internet, no Brasil, em relação a preços, assistência técnica, segurança e disponibilidade de serviços gratuitos.*
- *Você utiliza o computador por necessidade ou por pura escolha?.*
- *Se eu acho que Pascal realmente é a forma de programação que um Engenheiro Civil gostaria de aprender.*
- *Aonde utiliza o computador: em casa, na escola, no trabalho, etc.?*
- *Se a gente já tinha conhecimento básico prá cursar essa matéria?*
- *Sobre o custo operacional de um sistema educativo.*
- *Se as pessoas gostam de usar computadores ou são obrigadas?*
- *A informatização da vida humana não levará acarretará a uma falta de relacionamento humano?*
- *Uma pergunta discursiva do tipo: qual a importância do computador na sua vida?*
- *Quais os conhecimentos e em que nível está o entrevistado?*
- *Você possui computador em casa que possa ser utilizado a qualquer hora?*
- *Você tem facilidade de acesso a computadores?*

6.3 Pesquisa com usuários de um ambiente para Ensino de ICC

Nesta seção são mostrados os resultados obtidos quando os alunos utilizaram o ambiente descrito na seção 5.13, para Ensino de conceitos introdutórios de Computação para os Cursos de Agronomia e Engenharia de Aqüicultura no CCA. Para documentação da pesquisa, ao final da disciplina, os alunos responderam um questionário de opiniões (*Opinário 0*). Este opinário continha cerca de 20 perguntas. A coleta destes dados foi feita em uma turma de apenas 41 alunos, pois o objetivo maior era detectar o funcionamento correto do ambiente para Ensino e auxiliar na construção do opinário que foi mostrado na seção 6.2.

Para facilitar o entendimento e a visualização, as perguntas foram aqui transcritas e alguns comentários foram feitos quanto aos resultados obtidos. Em relação às respostas foram colocados as porcentagens ao lado de cada de uma das respostas.

1. *Você já utilizou o computador anteriormente ?*

- *Sim - (95 %)*
- *Não - (5 %)*

Esta questão visava saber se opinário era aplicável ao aluno. Nota-se que a maioria dos alunos já havia utilizado o computador anteriormente.

2. *Se sim, há quanto tempo utiliza o computador?*

- *Menos de 6 meses - (5 %)*
- *1 ano - (18 %)*
- *Mais de 2 anos - (77 %)*

A intenção desta questão era saber qual o grau de familiaridade existente com o computador. Os alunos já estavam bem familiarizados com o mesmo pois a maioria respondeu que utiliza o computador há mais de 2 anos.

3. *Atualmente, quantos dias num mês você utiliza o computador?*

- *Menos de 5 dias (15 %)*
- *De 10 a 15 dias (41 %)*
- *Mais de 15 dias (44 %)*

A questão 2 procurou identificar o nível do usuário (iniciante, médio ou avançado) e é complementada pelas perguntas 3, 4 e 5. Ao iniciar a disciplina a maioria declarou que o uso do computador há mais de 2 anos. Entretanto, as opiniões estão divididas no uso do computador de 10 a 15 dias no mês, o que pode ser considerado como uma característica de uso não exclusivo mas constante.

4. *Quantas horas costuma utilizar o computador diariamente?*

- *Menos de 1 hora (56 %)*
- *2 a 4 horas (34 %)*
- *Mais de 4 horas (10 %)*

Mais da metade dos alunos responderam que utilizavam o computador menos de 1 hora por dia.

5. Marque com um X as principais utilizações que faz do computador?

- () *Digitar textos.* (75 %)
- () *Desenhar figuras ou manipular fotos.* (24 %)
- () *Jogar.* (56 %)
- () *Escutar música.* (44 %)
- () *Assistir filmes e clipes.* (7 %)
- () *Navegar na Internet (pesquisas, entretenimento, informações gerais).* (95 %)
- () *Ler e enviar mensagens eletrônicas.* (75 %)
- () *Usar programas de comunicação na Internet ("bate-papos", conferências).* (27 %)
- () *Efetuar cálculos (planilhas, simulações, etc.).* (27 %)
- () *Aprender utilizar programas (educativos, de uso geral, etc.)* (27 %)
- () *Construir seus próprios programas.* (2 %)
- () *Outros* (2 %)

As opções da questão 5 visavam identificar as preferências dos usuários quanto ao uso que fazem do computador. Algumas perguntas visavam descobrir se o aluno possuía conhecimentos técnicos e a nível de saber programar a máquina.

Os principais usos do computador eram: navegar na rede, enviar mensagens eletrônicas e digitar textos. Uma parcela significativa utilizava o computador para entretenimento (escutar música, jogar, assistir filmes, etc.).

6. Se estiver no Laboratório, (fora do horário de aulas), em casa ou na empresa em que trabalha e, ao manipular o computador o mesmo paralisa, enviando uma mensagem de erro que julgue incapaz de compreender o que ocorreu, enumere as ações que geralmente toma em ordem crescente de prioridade para resolver tal problema.

- () *Tenta resolver sozinho.*
- () *Utiliza a "ajuda" do programa, caso seja possível.*
- () *Pergunta a alguém (amigo, colega, professor, etc.) que esteja perto dependendo do local onde você esteja.*
- () *Recorre a um livro ou manual do programa ou mesmo do computador.*

() *Chama a assistência técnica ou um profissional da computação.*

() *Se irrita, desiste e "abandona furiosamente" o computador.*

A questão 6 pretendia identificar o grau de iniciativa e independência do usuário ao lidar com a máquina. Verificou-se que as seqüências de ações mais tomadas eram: resolver o problema sozinho, em seguida pedir auxílio a alguém, recorrer ao manual, chamar a assistência técnica e somente no último caso abandonar o computador furiosamente.

7. *O que você acha que significa na computação o termo "von Neumann". Marque com X as opções que julgar mais provável:*

() *Um tipo de computador muito moderno. (0 %)*

() *Um componente eletrônico do computador inventado por John von Neumann. (7 %)*

() *O nome de um fabricante de computador. (2 %)*

() *O nome de um inventor de um sistema operacional. (2 %)*

() *Nenhuma das anteriores. (2 %)*

() *Desconhece o termo. (85 %)*

A questão 7 procura identificar usuários com conhecimentos avançados por ser bem específica da área da Computação. Como esperado, a maioria dos alunos (apenas usuários do computador) foi sincera respondendo que "desconhece o termo 'Von Neumann'". Notou-se que alguns alunos foram sugestionados pela resposta "um componente eletrônico do computador inventado por John von Neumann". Somente uma pequena parcela respondeu "nenhuma das anteriores" (que seria a resposta correta) o que não garante que estas respostas sejam conscientes para que seja possível identificar o nível de conhecimento avançado entre esses alunos. Comparando os resultados da avaliação da disciplina e com a interação feita pela autora com os alunos, foi possível identificar somente um usuário avançado nesta população. O resultado satisfatório nesta questão é admitir que a maioria foi sincera e que o perfil geral não são de usuários que possuem conceitos avançados.

8. *O termo "laço" num programa refere-se a:*

() *A união de um programa a outro em forma de "laço". (5 %)*

() *Um tipo de estrutura de controle utilizada na programação de computadores. (23 %)*

- () *Uma mensagem de erro que o computador emite quando ocorre um erro, fazendo com que o mesmo volte a seu estado anterior. (10 %)*
- () *Desconhece o termo. (62 %)*

Outra pergunta com a intenção de identificar o nível do usuário é a questão 8. Nesta questão, propositalmente, os alunos foram “induzidos” a fornecer respostas erradas como “a união de um programa a outro em forma de laço” e “uma mensagem de erro que o computador emite quando ocorre um erro, fazendo com que o mesmo volte a seu estado anterior”. Na Análise de Oportunidades é interessante utilizar estes tipos de recursos para detectar o ângulo do aluno. Estas informações podem ser úteis na descoberta de respostas para as perguntas do tipo “o que o cliente quer?” ou “o que o cliente sente?” na concepção do programa educacional.

9. *Utiliza muito os “assistentes” que auxiliam em tarefas tais como: elaborar um currículo, escrever uma página para Internet, enviar um Fax, conectar a Internet, instalar programas, resolver problemas de configuração de programas no computador.*

- () *Com frequência (15 %)*
- () *Às vezes (20 %)*
- () *Raramente - (65 %)*

Na questão 9, verificou-se que o uso de assistentes era uma prática comum entre usuários mais avançados.

10. *Você está cursando uma disciplina de fundamentos de computação. O material disponibilizado na rede é um apoio didático a esta disciplina. O computador também tem sido usado como coadjuvante no ensino geral. Você acha que o ensino com computador numa disciplina é melhor do que o ensino sem computador?*

- () *Melhor ensino COM computador (90 %)*
- () *Melhor ensino SEM computador (7 %)*
- () *Omissos - (3 %)*

Na questão 10, a maioria parece unânime na utilização do computador como apoio educacional, entretanto existem os casos omissos. Conforme será mostrado na seção 6.2 essas opiniões sofreram mudanças no decorrer da disciplina.

11. *Você acha que seria melhor que o material apresentado em forma de hipertexto fosse apresentado na forma de um livro convencional? Note que num livro, uma rápida “folheada” permite visualizar a existência de figuras, a extensão do livro, etc. Assim, você tem uma noção da quantidade de material que tem pela frente, o que não acontece com o ensino com computador na forma de hipertexto.*

Livro convencional (66 %)

Hipertexto (24 %)

Omissos (10 %)

Nota-se que na questão 11, a maioria, apesar de responder na questão 10 que o Ensino com computador numa disciplina é melhor do que o ensino sem computador, ainda prefere o livro convencional.

12. *Muitas vezes, na navegação entre páginas de um hipertexto na rede são encontradas ligações e páginas inacessíveis. Ou seja, ao passar de uma ligação a outra ou mesmo entre páginas encontra-se alguma mensagem de “erro” ou de “página não encontrada”. Geralmente, estes erros podem ser oriundos de problemas na rede que podem tornar as páginas inacessíveis e outras vezes são causados pela má elaboração do hipertexto que conterà ligações erradas ou inexistentes. Você encontrou estes tipos de erros no material disponibilizado na Internet para auxílio a esta disciplina?*

Diversas vezes (15 %)

Às vezes (24 %)

Raramente (15 %)

Nunca (46 %)

A questão 12 está relacionada com os aspectos construtivos do programa testado. Na modelagem do grafo de implementação do sistema considerou-se que todos estados do hipertexto fossem alcançáveis, para evitar erros de “páginas não encontradas”. Entretanto, deve ser observado a limitação do servidor da rede e das máquinas utilizadas no laboratório do CCA. Este tipo de questão é mais adequada na verificação de erros e testes do sistema.

13. *O que você acha mais interessante, fazer exercícios de múltipla escolha em material impresso ou utilizar páginas na Internet contendo exercícios (com opções de múltiplas escolhas). Supondo-se que nas páginas da Internet, ao final de cada exercício você saberá se escolheu a resposta certa ou não.*

- *Exercícios - material impresso (34 %)*
- *Exercícios - material "on-line" via Internet (66 %)*

A questão 13 parece indicar uma contradição com a questão 11 na qual os alunos apontam preferências para o material impresso. Entretanto, isto pode indicar um desejo de um maior dinamismo na avaliação de seu próprio conhecimento, nas aulas, na exploração e na fixação do conteúdo.

14. *Após utilizar o material apresentado na rede, você acha que o computador auxilia no processo de ensino? Marque as opções que considera pertinentes a esta questão.*

- () O conteúdo é apresentado numa forma mais atrativa. (24 %)*
- () A utilização do computador facilita o processo de aprendizagem porque o torna este processo mais rápido. (24 %)*
- () As ações de navegação podem ser repetidas várias vezes até a fixação do conteúdo. (32 %)*
- () O computador não deve servir de apoio a disciplinas pois é algo complicado de se manipular. (0 %)*
- () Nunca utilizo o material para tirar dúvidas sobre algum conceito. (8 %)*
- () Nenhuma das opções. (12 %)*

Na questão 14 a maioria respondeu que as ações de navegação auxiliam na fixação do conteúdo e que a utilização do computador torna este processo mais rápido e mais atrativo. A inclusão de sons, imagens, animações, etc., auxiliam muito o processo de entendimento e retenção dos conteúdos abordados em sistemas de ensino baseados em hipertexto.

15. *Você acha que a aprendizagem por computador é mais conveniente pois quando se cansa de manipular o computador desliga-o e só retorna quando tem vontade. Acha que é então que este processo é menos fatigante?*

- () Sim (39 %)*
- () Não (54 %)*
- () Omissos (7 %)*

Na questão 15 a maioria discorda que seja um processo menos fatigante. Em parte isto também pode ser entendido como uma certa resistência dos alunos

(principalmente de áreas que não sejam de computação) quanto as condições da aprendizagem (máquinas lentas e obsoletas, problemas de conectividade, etc.) que dificultam o processo de aprendizagem, desmotivando o estudante.

16. *Para aprender você prefere ler o material diretamente na tela do computador ou prefere imprimir o que está na tela no papel? Note que a utilização do computador você tem as facilidades das ligações do hipertexto.*

- () Ler diretamente na tela (32 %)*
- () Imprimir o material para leitura (68 %)*

Na questão 16, mais uma vez os alunos afirmam a preferência por material impresso como na questão 11. Mas isto entra em conflito com a questão 13 que mostra a preferência de exercícios “on-line”.

17. *Alguma vez, você quando procurava uma determinada informação em uma página, entrou em outra e não soube mais como retornar ao ponto anterior, ou perdeu-se no ponto em que estava e teve que iniciar toda navegação novamente?*

- () Sempre (2 %)*
- () Às vezes (19 %)*
- () Raramente (54 %)*
- () Nunca (25 %)*

Na questão 17 uma parte significativa dos estudantes afirma que gostaria de encontrar materiais semelhantes em outras disciplinas, assim como preferem a utilização do computador. Entretanto, isto é conflitante com a preferência dos mesmos de obter materiais impressos como abordado nas questões 11 e 16.

18. *Você gostaria de encontrar materiais semelhantes em outras disciplinas?*

- () Sim (78 %)*
- () Não (22 %)*

A questão 18 é semelhante a questão 12 e suas respostas estão coerentes com a mesma. Os problemas encontrados no material identificados por esta questão são os mesmos da questão 12, derivados do mal funcionamento da rede do CCA. Mais uma vez nota-se que a modelagem do grafo de implementação é importante para que não ocorram falhas construtivas no sistema.

19. *O acesso do material via rede (no laboratório ou através de uma conexão de acesso discado via modem conectado na linha telefônica) muitas vezes pode ser lento. A instalação das páginas localmente, no disco rígido do computador, eliminaria os problemas de lentidão e conectividade. Você acha que seria interessante ter um CD-ROM contendo o material, instalar num computador na sua casa ou escritório, para manipulá-lo a qualquer hora, independente das conexões da rede do laboratório ou de acesso discado? Por quê?*

() Sim (73 %)

() Não (27 %)

Quando perguntados, na questão 19, sobre o uso de um disco compacto (CD-ROM) contendo o material, a maioria afirmou que o acesso seria mais fácil, econômico e cômodo. Dentre as justificativas mais encontradas nesta preferência citam-se:

• *Alunos favoráveis ao uso do material em CD-ROM:*

- “... acesso ao material sem necessidade de Internet;*
- comodidade;*
- não haveria problemas de tempo de conexão, o material poderia ser acessado várias vezes;*
- dispensaria gastos com telefone/provedor ;*
- não estaria sujeito a flutuações da rede telefônica, páginas inacessíveis;*
- perderia menos tempo, teria página com acesso mais rápido;*
- Internet nem sempre funciona;*
- Além de evitar gasto telefônico é mais prático pois evita os pequenos transtornos que a Internet nos trás às vezes;*
- Desse modo os problemas diminuiriam;*
- Tendo o material em mãos não se necessita estar conectado o que diminui além de tempo de utilização também o gasto com a Internet;*
- Ter um CD-ROM facilita ao invés de ficar baixando aulas pela Internet;*
- Para fazer meu próprio estudo na hora que for conveniente;*
- Dispensaria a conexão e sua utilização é mais rápida;*
- Devido à rapidez obtida. Não corro o risco de ter linha ocupada ou lenta. Também porque existem pessoas que nem tem acesso à Internet;*

- *É mais prático, rápido e interessante;*
- *Mais rápido e acessível;*
- *Passaria para outras pessoas;*
- *Pela facilidade de acesso;*
- *Menor custo com telefone, mais rápido o acesso;*
- *Evitaria problemas de queda de conexão;*
- *Rapidez no acesso às páginas;*
- *O computador demora muito para abrir a página na Internet, com o CD-ROM seria mais rápido...”*
- Algumas respostas contrárias ao uso do material em disco compacto:
 - *“... Seria muito complicado;*
 - *Não tenho computador;*
 - *Pode ser difícil usar;*
 - *Não tenho CD-ROM no computador;*
 - *Teria mais coisa para estudar;*
 - *CD's são caros...”*

20. *Faça e críticas e sugestões sobre a “forma” do material apresentado:*

- Alunos que fizeram críticas (54 %)
- Omissos (46 %)

Na questão 20 os alunos fizeram comentários, críticas e sugestões. Dentre os principais comentários tem-se:

- *“... A forma como foi apresentado foi boa mas o problema é que tem muito conteúdo;*
- *A respeito do material disponível na Internet achei interessante, sendo que posso visualizar em casa, no trabalho, etc. No entanto, o material é muito extenso, poderia ser um pouco mais resumido e mais objetivo;*
- *Muito básico;*
- *O ensino com computador é muito cansativo;*
- *Faltou mais atividades práticas;*
- *Prático e interessante;*

- *Acho que o material está bom. Porém, é totalmente impossível estudar para a prova sem imprimir a matéria porque ninguém agüenta ficar lendo direto em frente da tela do computador que faz mal para os olhos;*
- *Que se façam mais tópicos e não assuntos longos. Creio que mais links agilizariam o processo;*
- *Deve começar desde o início para que o aluno se ache melhor dentro do sistema sem ficar por fora, deve ser mais organizado. Isto é, os tópicos foram dados na sala de aula estão em ordem diferente do que está no material;*
- *O material é bem escrito e apresentado. Em contrapartida, ainda não me acostumei a deixar de lado o lápis e o papel;*
- *O material é apresentado de forma acessível;*
- *Com certeza é uma forma de auxílio ao estudo obtido dentro da sala de aula mas não seria tão eficiente se fosse apresentado como único recurso para aprendizagem;*
- *Infelizmente, nem tudo dá para copiar e imprimir em casa;*
- *O material apresentado está de forma clara, porém as conexões que faço de casa por dia aumentam minha conta telefônica;*
- *O computador é um bicho de sete cabeças, não sei o que dizer;*
- *O material está bem apresentado, de fácil uso. No entanto, os exercícios poderiam ser realizados on-line...”*

Embora tenham ocorressem problemas nas máquinas e na rede do CCA, a manipulação do material foi satisfatória porque, segundo os dados obtidos, a modelagem do sistema garantiu todos os estados fossem alcançáveis. Todavia, observando-se as críticas, as sugestões e os comentários dos alunos, foram encontradas muitas respostas relacionadas a aceitabilidade do computador e ao Ensino como um todo.

As respostas do tipo “*pode ser difícil de usar*”, “*ninguém agüenta ficar lendo direto em frente da tela do computador*”, “*o ensino com computador é muito cansativo*” e “*o computador é um bicho de sete cabeças*” revelam uma resistência ao Ensino com computador. Entretanto, é interessante observar que respostas do tipo “*seria muito complicado*”, “*teria mais coisa para estudar*” e “*... o problema é que tem muito conteúdo*”, estão relacionadas não só com a aceitabilidade do computador mas aceitabilidade ao próprio Ensino universitário. E respostas do tipo “*CD’s são muito caros*” revelam o desinteresse do aluno pelo computador de um modo geral.

Uma observação pode ser feita em relação às turmas nas quais o Opinário 0 foi aplicado. São turmas do Curso de Ciências Agrárias cujos alunos não demonstraram interesse e/ou não estavam preparados para a computação.

Um exemplo mostra que se a computação não for ensinada de forma eficaz para a aplicação na área de interesse do aluno, este poderá achar que o computador não será útil.

Um dia, no Laboratório do CCA, a autora ouviu a seguinte frase de um aluno, cujo pai criava carpas na serra catarinense: “... *Professora, para que eu vou aprender informática? Eu vou mezer com terra, na roça ...*”.

Isto vem mostrar que o computador não é onipresente para toda a Sociedade. Muitas pessoas não gostam ou não se interessam por computadores. Cabe aos educadores e cientistas da computação, respeitadas as diferenças individuais, mostrar a utilidade do computador nas mais diversas áreas e nos mais diversos lugares do Brasil. Os conteúdos das disciplinas de ICC devem ser repensados.

Vale a pena ressaltar, que esta pesquisa foi feita em uma universidade de uma importante capital brasileira, com estudantes que não são da área de computação. A aceitabilidade do computador é um problema a ser resolvido por cientistas da computação mas a aceitabilidade ao próprio Ensino é um problema a ser resolvido pela Sociedade.

Vários erros na construção do opinário foram encontrados pela autora mas somente foram identificados após esta primeira pesquisa inicial. Dentre eles a identificação de casos omissos (alunos que não responderam determinadas questões), opções menos específicas (colocação da opção “outros”) questões de difícil avaliação (caso das seqüências de ações da questão 6) e, principalmente, a necessidade de um opinário inicial.

Aspectos	Respostas	Resultados
Utilização Anterior	Sim:	95
	Não:	5
Tempo utilização	Menos 6 meses:	5
	1 ano:	18
	Mais de 2 anos	77
Utilização Mensal	Menos 5 dias:	15
	10 a 15 dias:	41
	Mais 15 dias:	44
Utilização Diária	Menos de 1 hora:	56
	2 a 4 horas:	34
	Mais de 4 horas:	10
Principais Usos	Digitação:	75
	Navegação:	95
	E-mail:	75
	Entretenimento	56
Desconhece o termo	V. Neumann:	85
	Laço:	65
Utilização Assistentes	Com frequência:	15
	Às vezes:	20
	Raramente:	65
Ensino melhor	Com computador:	90
	Sem computador:	7
Livro ou Hipertexto?	Livro :	66
	Hipertexto:	24
	Omissos:	10
Exercícios	Impresso:	34
	"On-line":	66
Ensino menos fatigante ?	Sim:	39
	Não:	54
	Omissos:	7
Ler na tela ou imprimir?	Ler:	32
	Imprimir:	68
Favorável ao material local	Sim:	73
	Não:	27
Críticas e Sugestões	Opinaram:	54
	Não opinaram:	46

Tabela 6.3: Resultados no CCA

Capítulo 7

Epílogo

“Nam et ipsa scientia potestas est.”

Francis Bacon.

O histórico da evolução das máquinas mostram que as tentativas de aumento da rapidez, da eficiência do processo Ensino-aprendizado não é algo recente. O aprendizado de máquina pode ser organizado de forma a desenvolver o aprendizado humano. A formalização das questões relacionadas a possibilidade da máquina inteligente auxiliando o Ensino de seres humanos leva à reflexões sobre o próprio processo de Ensino. Segundo Brofferio [47], definir, projetar e implementar um sistema de Ensino envolve a resolução de três níveis de problemas inter-conectados: o nível tecnológico, o nível humano e organizacional e o nível econômico e legal. Neste caso, o hipertexto representou um grande passo no desenvolvimento dos computadores na educação. Inicialmente, o hipertexto não foi visto como um veículo adequado aos processos de Ensino-aprendizagem, mas o poder de acesso a informação oferecido por tais sistemas, favoreceu o projeto de ambientes de aprendizagem reativos e interativos, segundo Mayes [141]. Estes modelos procuravam sanar as dificuldades do diálogo dos sistemas com os estudantes. A hipermídia promoveu flexibilidade e aumentou o potencial destes sistemas. A rede mundial de computadores possibilitou um alto grau de interatividade com a manipulação direta dos conceitos através de uma grande base de dados de materiais de ensino hipermídia que podem ser disponibilizados.

Ensinar com o computador envolve o desenvolvimento de um programa. Normalmente, os programas educacionais não utilizam especificações formais. O desenvolvimento de programas é base da Engenharia de Programas. Os métodos formais

aplicados na construção de programas têm a grande vantagem de poderem ser utilizadas por várias pessoas distintas, sem grande variabilidade, devido ao fato de serem formais.

Esta tese apresentou a problemática da Engenharia de Programas Educacionais para Ensino via redes e uma proposta para modelagem construtiva de sistemas de ensino educacionais independente de máquinas, plataformas e domínios.

No desenvolvimento de um ambiente Hipermídia para Ensino-aprendizagem via rede, além dos aspectos psico-pedagógicos e da interface com o usuário, o programa necessita de especificação correta e precisa. Um ambiente de Ensino-aprendizado via rede de computadores, especificado corretamente, tem grandes chances de funcionar corretamente. Todavia, para o uso de Ensino via rede, algumas observações devem ser feitas em relação às estratégias de aprendizado. Para ensinar regras e manipulações de um determinado domínio, a apresentação virtual pode ser útil, mas não pode ser exclusiva porque deve haver uma interação com alunos. Além disso, as habilidades não podem ser ensinadas virtualmente. Para transmissão pura de conteúdos pode-se utilizar o aprendizado por decorar ou por ser contado, tendo-se menos esforço do que se fosse ensinado por descoberta. Por outro lado, para ensinar habilidades, é mais adequado o aprendizado por descoberta ou então por analogia. Isto leva a conclusões interessantes em relação ao ensino virtual e não virtual. O Ensino virtual ou ensino à distância é muito bom para transmitir o conhecimento utilizando a metodologia de ser contado ou de decorar. O ensino de manipular coisas deve ser um ensino que pode ser *auxiliado* por técnicas modernas de ensino à distância para dar exemplos e exercícios, mas deve haver uma parte de contato pessoal para que estes exemplos possam ser eficazes e identificados à luz de erros cometidos. Já o ensino de habilidade, que conduz a mudança no comportamento, tem que ser feito por descoberta. Para que o ensino seja realmente dinâmico, a participação do professor é fundamental pois, embora o computador auxilie a ampliação das capacidades intelectuais, ele não pode transmitir habilidades.

Em termos formais, este trabalho ressaltou, através da categorização de conceitos, que o problema da transmissão de conhecimentos pode ser atenuado com o uso de uma mídia mais rica. Não basta só a entrega de conteúdos por meio textual e o uso do computador pode sanar este problema pela possibilidade de inclusão de fala, música, sons, filmes, animações, etc. nos sistemas de ensino computadorizados. Com o uso do computador, o que se pretende é aumentar a eficiência do processo de ensino.

No Brasil são desconhecidas aplicações de métodos formais categóricos nos programas educacionais. O conceito de Ciclo de Vida em programas educacionais é um conceito novo que nasceu na Engenharia de Programas. O modelo Hipertômato

pode ser usado nesta abordagem definindo-se as fases de desenvolvimento do programa. Na primeira fase é feita a Análise de Oportunidades, na qual são verificadas as características relevantes do programa relacionadas com a finalidade deste. A Teoria das Categorias pode simplificar o processo de abstração e aplicação em diferentes domínios.

As implementações realizadas demonstram a potencialidade das ferramentas formais na concepção de ambientes de Ensino-aprendizagem computadorizados. Outras perspectivas para a continuidade deste trabalho poderão ser abordadas. É necessário ter uma teoria com um vocabulário que permita trabalhar abstratamente com objetos. A Teoria das Categorias possibilita, então, que o modelo categórico de Hipertômato seja uma ferramenta útil na especificação de programas com intuito de Ensino-aprendizagem. A flexibilidade desta modelagem formal é que qualquer estratégia pedagógica pode ser escolhida para o Ensino de um determinado domínio do conhecimento, mas para isso deve ser feita a Análise de Oportunidades e observada a aceitabilidade do computador através de um opinário.

O opinário permite extrair uma tabela de comparação de ensino com o material clássico e com o material não clássico. Os opinários permitiram identificar as preferências de alunos que não são da área de computação, além de identificar dados sobre o material apresentado para algumas turmas. Dessa maneira, pode-se pensar em seguir o Ciclo de Vida de um programa educacional. Os principais pontos abordados no opinário foram:

- Identificação do nível do usuário (iniciante, médio ou avançado) e das condições de uso do computador;
- Identificação das preferências dos usuários e preferências na utilização do computador
- como apoio educacional;
- Grau de iniciativa e de independência do usuário ao lidar com o computador;
- Casos omissos (alunos que não respondem determinadas questões);
- Preferências na utilização do computador como apoio educacional;
- Resultados esperados na modelagem do grafo de implementação do sistema, levando-se em consideração na sua construção que todos os estados do sistema hipermídia sejam alcançáveis;
- Identificação de Falhas construtivas no sistema.

Os resultados obtidos nas pesquisas de opinião motivam estudos futuros para o desenvolvimento de Análise de Oportunidades de ambientes de Ensino.

As perspectivas futuras deste trabalho é a utilização de métodos formais categóricos para desenvolvimento de programas de Ensino de um modo geral. Como é o primeiro trabalho nesta direção não se pretende alcançar a totalidade de pontos dos métodos formais mas pretende-se abrir um caminho novo:

Hipertômatos na Computação aplicada à Educação!

Referências Bibliográficas

- [1] ADÁMEK, J. “Automata and Categories: finiteness against minimality”. In: *Mathematical Foundations of Compute Science*, Lecture Notes in Computer Science, p. 160–166, Springer-Verlag, 1975.
- [2] AKHARAS, F. “From the Process of Instruction to the Process of Learning: Constructivist Implications for the Design of Intelligent Learning Environment”. URL: <http://www.cbl.leeds.ac.uk/~fabio/home.html>. Computer Based Learning Unit, University of Leeds, Leeds, England, 1996.
- [3] ALENCAR, E. *Teoria Elementar dos Conjuntos*. São Paulo : Livraria Nobel, 21 ed., 1990.
- [4] ALMEIDA, M. A. F. *Filtragem Digital de Sinais Biomédicos*. Dissertação de Mestrado, Programa de Pós-graduação em Engenharia Elétrica - UFSC, Florianópolis, Fevereiro, 1997.
- [5] ALMEIDA, M. A. F. *Aprender, atividade inteligente: e se esta inteligência for parcialmente artificial?* Dissertação de Mestrado, Curso de Pós-graduação em Ciência da Computação - UFSC, Florianópolis, Setembro, 1999.
- [6] ALMEIDA, M. A. F. “Hipertômatas na Computação aplicada à Educação”. Exame de Qualificação para Doutorado em Ciência da Computação, PGCC-INE-UFSC, Fevereiro, 2001.
- [7] ALMEIDA, M. A. F.; BARRETO, J. & LIMA, W. C. “Ensino de Filtragem Digital de Sinais Biomedicos Via Rede de Computadores”. In: *Primer Congreso Peruano de Ingeniería Biomédica - TUMI'99*, Pontificia Universidad Católica del Perú, Lima, Setiembre, 1999, p. 45–47.
- [8] ALMEIDA, M. A. F. & BARRETO, J. M. “Ferramenta Computacional para Filtragem de Sinais Biomédicos”. In: *Anais da 49ª Reunião Anual da SBPC*, Belo Horizonte, Julho, 1997, p. 49.

- [9] ALMEIDA, M. A. F. & BARRETO, J. M. “Filtragem não causal de sinais de urofluxometria”. In: *Anais da 49ª Reunião Anual da SBPC*, Belo Horizonte, Julho, 1997, p. 49. (A3-005).
- [10] ALMEIDA, M. A. F. & BARRETO, J. M. “Ambiente para Ensino do Projeto de Filtros Digitais no domínio do tempo: aplicação a sinais de eletrocardiografia”. In: *Anais do IV FNCTS - Fórum Nacional Ciência e Tecnologia em Saúde*, Curitiba, Paraná, Outubro, 1998, p. 407-408. (ISBN:85-7014-006-1).
- [11] ALMEIDA, M. A. F. & BARRETO, J. M. “Filtragem Digital de Sinais Biomédicos”. In: *Caderno de Engenharia Biomédica*, vol. 14, p. 94-94, Rio de Janeiro: Revista Brasileira de Engenharia - RBE, 1998. (ISSN: 0102-2644 - Resumo de Teses).
- [12] ALMEIDA, M. A. F. & BARRETO, J. M. “Filtragem não causal de sinais de urofluxometria”. In: *1er Congreso Latinoamericano de Ingeniería Biomédica*, Mazatlan, México, Noviembre, 1998, p. 729-732. (ISBN:968-5063-00-1).
- [13] ALMEIDA, M. A. F. & BARRETO, J. M. “Apoio ao ensino de Redes Neurais Artificiais via computador”. In: *XXII Congreso Argentino de Bioingeniería - SABI99*, Universidade de Favaloro, Argentina, Junho, 1999.
- [14] ALMEIDA, M. A. F. & BARRETO, J. M. “Formação continuada por Simulação em Ambiente Virtual usando Redes de Computadores”. In: *Pesquisa Naval - Suplemento Especial da Revista Marítima Brasileira*, n. 12, p. 271-287, Rio de Janeiro: Revista Marítima Brasileira, Outubro, 1999. (ISSN: 1414-8595).
- [15] ALMEIDA, M. A. F. & BARRETO, J. M. “Implementação de um Sistema Hiperfídia como Autômato para Ensino de Inteligência Artificial”. In: *Workshop de Teses e Dissertações em Informática em Educação do I Simpósio Catarinense de Computação*, UNIVALI, Agosto, 2000.
- [16] ALMEIDA, M. A. F. & BARRETO, J. M. “Modelagem de um Sistema Hiperfídia como Autômato para Ensino de Inteligência Artificial”. In: *Anais do XX Congresso Nacional da Sociedade Brasileira de Computação - SBC 2000*, Campus da PUC-PR, Curitiba, Julho, 2000, p. 1-8. (ISBN 85-7292-050-1).
- [17] ALMEIDA, M. A. F. & BARRETO, J. M. “Modelagem Categórica de Sistemas Hiperfídia como Autômato: Categoria dos Hipertômatos”. In: *Anais do XVI Simpósio Brasileiro de Sistemas Hiperfídia e Multímedia - SBMIDIA '2001*, Florianópolis, Outubro, 2001, p. 227-228.

- [18] ALMEIDA, M. A. F.; BARRETO, J. M. & CASTRO, A. C. R. "Non Causal filtering in Urofluxometry". In: *XVIII International Conference on Medical and Biological Engineering and XI International Conference on Medical Physics*, Nice, France, September, 1997, p. 549. (F83-PS1.04).
- [19] ALMEIDA, M. A. F.; BARRETO, J. M. & MACHADO, M. A. S. "Computação aplicada ao Ensino utilizando Inteligência Artificial". In: *Pesquisa Naval - Suplemento Especial da Revista Marítima Brasileira*, n. 14, p. 81-96, Rio de Janeiro: Revista Marítima Brasileira, Outubro, 2001. (ISSN: 1414-8595).
- [20] ALMEIDA, M. A. F.; BARRETO, J. M. & MACHADO, M. A. S. "Inteligência Artificial - uma aplicação ao Ensino". In: *Anais do Simpósio Brasileiro de Pesquisa Operacional, SOBRAPO'2001*, Campos do Jordão, 2001.
- [21] ALMEIDA, M. A. F.; BARRETO, J. M. & SANTOS, M. M. D. "Um ambiente Computacional para Ensino de Redes Neurais Artificiais". In: *Anais da 51ª Reunião Anual da SBPC*, Porto Alegre, Julho, 1999.
- [22] ALMEIDA, M. A. F.; BARRETO, S. Q.; CORRÊA, M. C. O. B. & SOUZA, L. L. "Aprendizado Cooperativo utilizando Construtivismo e Redes de Computadores no Ensino a Distância". In: *Anais da 51ª Reunião Anual da SBPC*, Porto Alegre, Julho, 1999.
- [23] ALSPECTOR, J. & ALLEN, R. B. "A Neuromorphic VLSI Learning System". In: *Advanced Research in VLSI: Proceedings of the 1987 Stanford Conference*, Cambridge, 1987, P.Losleben, Ed., MIT Press, p. 313-349.
- [24] ANTUNES, S. D.; BARRETO, J. M.; LEONARD, C. & MARCHAND, O. "PROFCOMP: Integrated Author System / Simulation Package". In: *MELECON'89 - IEEE Mediteranean Electrotechnical Conference*, Instituto Superior Técnico, Lisboa, April, 1989, p. 763-766.
- [25] APPLE. "Where Minds Meet". *Apple European University Consortium*. Amsterdam, 1989.
- [26] ARBIB, M. A. & MANES, E. *Arrows, Structures, and Functions - The Categorical Imperative*. New York : Academic Press, Inc., 1975.
- [27] ASPERTI, A. & LONGO, G. *Categories, Types, and Structures*. Foundations of Computing Series. Cambridge, Ma. : The MIT Press, 1991.
- [28] ÁVILA, L.; BRASIL, L. M. & ALMEIDA, M. A. F. "Ensino de redes neurais e seu uso como ferramenta auxiliar ao profissional da área da saúde". In: *SIARP'2001*, Florianópolis, Outubro, 2001.

- [29] ÁVILA, L.; BRASIL, L. M. & ALMEIDA, M. A. F. "Proposal of Actions for Creation Of One Multimedia Enviroment to a Tutorial In Artificial Neural Networks". In: *VJMEDIMAG'2001*, Vietnam, 2001.
- [30] BARR, A. & FEIGENBAUM, E. *Handbook of Artificial Intelligence*, vol. 2, cap. 9. Califórnia, USA: Heuristech Press, 1982.
- [31] BARRETO, J. M. "The Role of Bond Graphs in Qualitative Modeling". In: *12th IMACS World Congress on Scientific Computation*, Paris, July, 1988, p. 84-87.
- [32] BARRETO, J. M. "Lógica Nebulosa". (Notas de aulas - Mestrado em Engenharia Elétrica - GPEB-UFSC), 1995.
- [33] BARRETO, J. M. "Modelagem e Simulação de Sistemas Físicos e Biológicos". (Notas de Aulas - Mestrado em Engenharia Elétrica - GPEB-UFSC), 1995.
- [34] BARRETO, J. M. *Inteligência Artificial no Limiar do Século XXI*. Florianópolis, SC : Duplic, 1^ª ed., 1997.
- [35] BARRETO, J. M. "Introdução às Redes Neurais Artificiais". In: *V Escola Regional de Informática da SBC Regional Sul*, Santa Maria, Florianópolis, Maringá, Maio, 1997, p. 41-71.
- [36] BARRETO, J. M. *Inteligência Artificial no Limiar do Século XXI*. Florianópolis, SC : Duplic, 2^ª ed., 1999.
- [37] BARRETO, J. M. *Inteligência Artificial no Limiar do Século XXI*. Florianópolis, SC : Duplic, 3^ª ed., 2001.
- [38] BARRETO, J. M. *Teoria da Computação*. Florianópolis, SC : Preprint, 2001.
- [39] BARRETO, J. M. & LEFÈVRE, J. "The Physiology of a simulation program". In: *Anais do IV Encontro Nacional de Automática*, Florianópolis, Julho, 1983.
- [40] BARRETO, J. M.; NEYER, M. D.; LEFÈVRE, P. & GOREZ, R. "Qualitative Physics versus fuzzy Sets theory in Modeling and Control". In: *IECON'91: IEEE International Conference on Industrial Electronics, Control and Instrumentation*, Kobe, Japan, October, 1991, p. 1651-1656.
- [41] BARROS, L. A. & DA SILVA BORGES, M. R. "ARCOO - Sistema de Apoio à Aprendizagem Cooperativa Distribuída". In: *VI Simpósio Brasileiro de Informática na Educação*, Florianópolis, SC, 1995, p. 88-100.

- [42] BARRY, D. & STANIENDA, T. "Solving the JAVA Storage Problem". *IEEE Computer*, n. 11, p. 12-26, November, 1998.
- [43] BAUDEL, B. & CANTEGRIT, E. "Smalltalk and Simulation of Batch-processes". In: *Artificial Intelligence, Expert Systems and Languages in Modeling and Simulation*, p. 295-300, North-Holland: Elsevier Science Publishers B. V., 1988.
- [44] BORDELEAU, P. "*Les développements technologiques:l'univers des environnements pédagogiques informatisés virtuels*". Montréal : Éditions Logiques, 1^a ed., 1994.
- [45] BORDELEAU, P. "L'histoire des Technologies Informatiques et quelques-unes de leurs applications en éducation". Faculté des sciences de l'éducation, Université de Montréal, 1999.
- [46] BRENT, W. "*Reflections on Constructivism and Instructional Design*", vol. 2, p. 12-21. Englewood Cliffs, NJ: Educational Technology Publications, 1997.
- [47] BROFERIO, S. C. A. "University Distance Lesson System". *IEEE Transactions on Education*, n. 41, p. 17-24, February, 1998.
- [48] BUCHANAN, B. G. "Brief History of Artificial Intelligence by Bruce Buchanan". URL: <http://www.cs.pitt.edu/~peterson/html/bbhist.html>. University of Pittsburgh, 1998.
- [49] BÜRLE, G. "The role of qualitative reasoning in modeling". In: *Artificial Intelligence, Expert Systems and Languages in Modeling and Simulation*, p. 57-62, North-Holland: Elsevier Science Publishers B. V., 1988.
- [50] CAMERON, J. R. "*Tutorial: JSP & JSD: The Jackson Approach to Software Development*". Silver Spring : IEEE Computer Society, 1976.
- [51] CARBONELL, J. "AI in CAI-An Artificial Intelligence Approach to Computer Assisted Instruction". *IEEE Transaction on Man-Machine Systems*, n. 11, p. 190-202, 1998.
- [52] CARBONELL, J. G.; MICHALSKI, R. S. & MITCHELL, T. "*An Overview of Machine Learning*", cap. 1, p. 3-23 : Springer-Verlag, 1984.
- [53] CARD, H. & MOORE, W. R. VLSI circuits for neural networks. *International Journal of Neural Networks Systems*, n. 2, p. 149-165, 1989.

- [54] CARDOSO, L. N.; ALMEIDA, M. A. F. & STEIN, M. G. F. "Tutorial hipertexto sobre o manuseio do livro didático". In: *Anais da 5ª Reunião Especial da SBPC*, Blumenau, Setembro, 1997, p. 207. (B-6.024).
- [55] CARLSON, B.; BURGESS, A. & MILLER, C. "Timeline of Computing History". URL: <http://www.computer.org/computer/timeline/>. The Institute of Electrical and Electronics Engineers, Inc., 1996.
- [56] CHOU, C. "Construction of a Virtual Reality Learning Environment for teaching structural analysis". In: *Computer Applications in Engineering Education*, n. 4, New York: Wiley-Interscience, 1997.
- [57] CLANCEY, W. "*Transfer of rule-based expertise through a tutorial dialog*". Tese de Doutorado, Computer Science Department, Stanford University, (STAN-CS-769), Stanford, 1979.
- [58] CLANCEY, W. "From GUIDON to NEOMYCIN and HERACLES in Twenty Short Lessons". In: *Current Issues in Expert Systems*, p. 79-124, London: Academic Press, 1987.
- [59] CLANCEY, W. "Intelligent Tutoring Systems: A Survey". In: *Current Issues in Expert Systems*, p. 39-78, London: Academic Press, 1987.
- [60] CLANCEY, W. "The role of qualitative models in instruction". In: *Artificial intelligence and human learning: intelligent computer-aided instruction*, p. 49-68, London: Chapman and Hall, 1988.
- [61] CLANCEY, W. J. *Knowledge-Based Tutoring - The GUIDON Program*. Cambridge, Massachusetts : The MIT Press, 1987.
- [62] COSTELLA, A. *Comunicação - Do Grito ao Satélite*. São Paulo : Editora Mantiqueira, 1984.
- [63] DAYHOFF, J. E. *Neural Networks Architectures*. New York, USA : Van Nostrand Reinhold, 1992.
- [64] DEDEN, A. "Computers and Systemic Change in Higher Education". *Communications of the ACM*, n. 41, p. 58-63, January, 1998.
- [65] DIVERIO, T. A. & MENEZES, P. B. *Teoria da Computação - Máquinas Universais e Computabilidade*. Porto Alegre : Sagra Luzzato, 1999.

- [66] DOPPKE, J. C. & HEIMBGNER, D. "Software Process Modeling and Execution within Virtual Environments". *ACM Transactions on Software Engineering and Methodology*, n. 7, p. 1–40, January, 1998.
- [67] DOS SANTOS, M. A. P. "Computadores na Educação". (Projeto de Tese - Mestrado em Educação - PUC-RJ), Julho, 1987.
- [68] DOUKIDIS, G. I. "Using LISP for developing discret events simulations models". In: *Artificial Intelligence, Expert Systems and Languages in Modelling and Simulation - IMACS*, p. 105–110, North-Holland: Elsevier Science Publishers B. V., 1988.
- [69] DRESCHER, G. Genetic ai - translating piaget into lisp. Relatório Técnico 890, Massachusetts Institute of Technology - Artificial Intelligence Laboratory, Massachusetts, USA, February, 1986. (AI-MEMO).
- [70] DREYFUS, H. L. *What Computers can't do*. Cambridge, Massachusetts : MIT Press, 1972.
- [71] DREYFUS, H. L. *What computers still can't do*. Cambridge, Massachusetts : MIT Press, 1992.
- [72] DUMM, T. "An Introduction to the Science of Artificial Intelligence by Tim Dumm". URL: <http://tqd.advanced.org/2705/index.html>, 1997.
- [73] DURR, E. & VAN KATWIJK, J. "VDM++ - A Formal Specification Language for Object-oriented Designs". In: *Computer Systems and Software Engineering Proceedings of CompEuro '92*, 1992, IEEE Computer Society Press, p. 214–219.
- [74] EL-REWINI, H. & MULDER, M. C. "Keeping Pace with an Information Society". *IEEE Computer Innovative technology for computer professionals*, n. 11, p. 46–47, November, 1997.
- [75] EUCLIDES. *The Elements*, vol. 1. New York : Dover, 1956.
- [76] EUCLIDES. *The Elements*, vol. 2. New York : Dover, 1956.
- [77] FERNANDES, C. T. & SANTOS, N. "Pesquisa e Desenvolvimento em Informática na Educação no Brasil - Parte I". *Revista Brasileira de Informática na Educação - RBIE*, n. 4, p. 9–39, Abril, 1999.
- [78] FERNANDES, C. T. & SANTOS, N. "Pesquisa e Desenvolvimento em Informática na Educação no Brasil - Parte II". *Revista Brasileira de Informática na Educação - RBIE*, n. 5, p. 19–39, Julho, 1999.

- [79] FERREIRA, A. B. H. *Novo Dicionário Aurélio Século XXI*. Nova Fronteira, 2000.
- [80] FIADEIRO, J. & MAIBAUM, T. "A mathematical toolbox for the software architect". In: *Proceedings of the 8th International Workshop on Software Specification and Design*, 1996, p. 46-55.
- [81] FIALHO, F. A. P. "As Atividades Cognitivas - Uma Introdução à Engenharia do Conhecimento". Programa de Pós-graduação em Engenharia de Produção, UFSC, 1998.
- [82] FINLAY, J. *Software Personalization*, p. 1240-1241. New York: Van Nostrand Reinhold, 1993.
- [83] FISHWICK, P. A. "Web-Based Simulation: Some Personal Observations". *1996 Winter Simulation Conference*, n. 1, p. 772-778, December, 1996.
- [84] FLEISCHHAUER, L. I. A.; BARBOSA, A. C. G.; VAVASSORI, F. B. & GAUTHIER, F. A. O. "A experiência de Criação de um Curso através de Software de Autoração de Multimídia: Uma Introdução". In: *ANAIS do III Congresso Argentino de Ciência de la Computacion (CACIC)*, La Plata, Argentina, 1997, vol. 2, p. 812-821.
- [85] FODOR, J. A. *Concepts where Cognitive Science went wrong*. New York : Clarendon Press - Oxford, 1998.
- [86] FOGEL, D. B. "Genetic Algorithms". *Scientific American*, n. 267, p. 66-72, July, 1992.
- [87] FOGEL, D. B. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. IEEE Press, 1995.
- [88] FOKKINGA, M. M. "Calculate Categorically!". *Formal Aspects of Computing*, n. 4, p. 673-692, 1992.
- [89] FORBUS, K. "Qualitative process theory". *Artificial Intelligence*, n. 24, p. 85-168, 1984.
- [90] FORESTER, T. & MORRISON, P. *Computer Ethics*. Massachusetts : The MIT Press, 3ª ed., 1995.
- [91] FREEMAN, J. A. & SKAPURA, D. M. *Neural Networks: algorithms, applications, and programming techniques*. USA : Adisson-Wesley Publishing Company, Inc., 1992.

- [92] FREIRE, P. *Educação como prática da liberdade*. Rio de Janeiro, RJ : Editora Paz e Terra, 1986.
- [93] FRIEDMAN, N. & HAFNER, C. "The State of the Art in Ontology Design: A survey and Comparative Review". *AI Magazine*, n. 18, p. 53-74, 1997.
- [94] FUTO, I. H. "AI and Simulation Prolog basis". In: *Artificial Intelligence, Expert Systems and Languages in Modelling and Simulation, IMACS*, p. 15-20, North-Holland: Elsevier Science Publishers B. V., 1988.
- [95] GALLANT, S. "Connectionist Expert Systems". *Communications of the ACM*, n. 31, 1988.
- [96] GALLANT, S. I. *Neural network learning and expert systems*. Cambridge, Massachussets : MIT press, 1993.
- [97] GIL, S. Q.; SOUZA, L.; FIALHO, F. P. & ALMEIDA, M. A. F. "A vida pede licença para entrar na Escola". In: *Anais da 49ª Reunião Anual da SBPC*, Belo Horizonte, Julho, 1997. (B-6.192).
- [98] GOGUEN, J. A. "A Categorical Manifesto". *Mathematical Structures in Computer Science*, n. 11, p. 49-67, 1991.
- [99] GOLDEN, R. M. *Mathematical Methods for Neural Network Analysis and Design*. Massachussets : The MIT Press, 1996.
- [100] GOOD, T. L. & BROPHY, J. E. *Educational Psychology: a realistic approach*. New York : White Plains, Longman, 4 ed., 1990.
- [101] GOODWING, N. C. "Funcionalidade and Usability". *Communications of the ACM*, n. 30, p. 229-233, March, 1987.
- [102] GORDON, P. "The self-organising teacher". In: *Automated Teaching Bulletin*, vol. 1, The Rheem-Califone Corp., December, 1959.
- [103] GROUP, T. R. L. *The RAISE Specification Language*. Prentice Hall BCS Practitioner Series, 1992. ISBN 0-13-752833-7.
- [104] GUNAWARDENA, J. *New Connections between Mathematics and Computer Science: report*. Relatório Técnico 96002, Hewlett Packard Labs, 1996.
- [105] GURR, C. A. & TOURLAS, K. "Towards the principled design of software engineering diagrams". In: *International Conference on Software Engineering*, 2000, p. 509-518.

- [106] HARGER, R. O. "Introducing DSP with an Eletronic Book in Computer Classroom". *IEEE Transactions on Education*, n. 39, p. 173-179, May, 1996.
- [107] HERGENHAHN, B. R. *An Introduction to Theories of Learning*. Englewood Cliffs, N. J. : Prentice-Hall, Inc., 1982.
- [108] HICKMAN, F. R.; KILLIN, J. L.; LAND, L.; MULHALL, T.; PORTER, D. & TAYLOR, R. M. *Analysis for Knowledge based-systems: a practical guide to KADS methodology*. London : Ellis Horwood Limited, 1989.
- [109] HOARE, C. "Notes on an Approach to Category Theory for Computer Scientists". In: *Constructive Methods in Computing Science*, 1989, M. Broy, Ed., vol. 55, International Summer School, Springer Verlag, p. 245-305.
- [110] HOFSTEDE, A. H. M.; LIPPE, E. & FREDERIKS, P. J. M. "Conceptual Data Modeling from a Categorical Perspective. *The Computer Journal*, n. 39, p. 215-231, 1996.
- [111] HOFSTEDE, A. H. M. & PROPER, H. A. "How to Formalize It? Formalization Principles for Information System Development Methods". Relatório Técnico FIT-TR-1997-04, October, 1997.
- [112] HOLLAND, J. H. *Adaptation in natural and artificial systems*. Ann Arbor, MI : The University of Michigan Press, 1981.
- [113] HOPCROFT, J. E. & ULLMAN, J. D. *Introduction to automata theory, languages, and computation*. New York : Addison-Wesley Publishing Company, Inc., 1979.
- [114] IBM. "The Aglet Workbench". <http://www.trl.ibm.co.jp/aglets>, 1999.
- [115] IWASAKI, Y. & SIMON, H. A. "Causality in device Behavior". *Artificial Intelligence*, n. 29, p. 3-31, 1986.
- [116] IWASAKI, Y. & SIMON, H. A. "Theories of Causal Ordering: reply". *Artificial Intelligence*, n. 29, p. 63-72, 1986.
- [117] JACKSON, M. A. *Construtive Methods of Program*, cap. 2, p. 67-99. A Monograph in The Computer Society Press : IEEE Computer Society, 1976. (Reprints).
- [118] JOHANSSON, M.; GÄFVERT, M. & ASTRÖM, K. J. "Interactive Tools for Education in Automatic Control". *IEEE Control System - Interactive Learning*, n. 18, p. 33-40, June, 1998.

- [119] KAPLAN, E. "History of Calculating Machines - Introduction". URL: <http://www.webcom.com/calc/>. 1996.
- [120] KAPLAN, R. & ROCK, D. "New directions for intelligent tutoring". *Artificial Intelligence Expert*, n. 5, p. 31-40, February, 1995.
- [121] KASABOV, N. K. *Foundations of neural networks, fuzzy systems, and knowledge engineering*. Massachusetts, USA : The MIT Press, 1996.
- [122] KINIRY, J. & ZIMMERMANN, D. "A hands-on look at Java Mobile agents". *IEEE - The Internet Computing*, n. 4, p. 21-30, August, 1999.
- [123] KLEER, J. D. "How circuit work". *Artificial Intelligence*, n. 24, p. 169-204, 1984.
- [124] KLEER, J. D. & BROWN, J. "A qualitative physics based on confluences". *Artificial Intelligence*, n. 24, p. 7-83, 1984.
- [125] KNOOP, J.; KOSCHUETZKI, D. & STEFFEN, B. "Basic-block graphs: living dinosaurs?". *Lecture Notes in Computer Science*, n. 1383, p. 65-87, 1998.
- [126] KREUTZ, L. S. "Sistema de Ensino de Fisiologia Cardíaca - INE - UFSC". URL: <http://www.inf.ufsc.br/~schuch>, 1998. (Departamento de Informática e de Estatística - INE-UFSC).
- [127] KRÖSE, B. J. A. & VAN DER SMAGT, P. P. *An Introduction to Neural Networks*. University of Amsterdam, 5^a ed., January, 1993.
- [128] KUIPERS, B. "Commonsense Reasoning About Casuality: Deriving Behavior from Structure". *Artificial Intelligence*, n. 24, p. 169-204, 1984.
- [129] KUIPERS, B. "Qualitative Simulation". *Artificial Intelligence*, n. 29, p. 229-338, 1986.
- [130] LAMBEK, J. & SCOTT, P. J. *Introduction to higher order categorical logic*. Cambridge : Cambridge University Press, 1986. (Cambridge Studies in Advanced Mathematics 7).
- [131] LEFÈVRE, J. & BARRETO, J. M. "Didatic microcomputer simulation in cardiac dynamics". In: *5 th Annual Conference on Frontiers of Engineering in Health Care (IEEE)*, Columbus, Ohio, EUA, 1983. (Resumo no IEEE Trans. Biomedical Engineering, v. BME-3,0 p. 512).

- [132] LINDEN, E. "O que os animais realmente pensam?". In: *Revista Seleções*, C. Kierchel, Ed., p. 92-97, Reader's Digest Brasil Ltda., Fevereiro, 2001.
- [133] MAHAJAN, R. & SHNEIDERMAN, M. "Visual and Textual Consistency Checking Tools for Graphical User Interfaces". *IEEE Transactions on Software Engineering*, n. 23, p. 722-735, 1997.
- [134] MAMROUD, Y. "Artificial Intelligence - History, Philosophy and Practice". URL: <http://royce-ap.com/ai/links.html>, 1999. (Philosophy Department, Tel Aviv University).
- [135] MANCORIDIS, S.; HOLT, R. & GODFREY, M. "Tool support for software engineering education". In: *Proceedings of the ICSE-16 Workshop on Software Engineering*, 1994, p. 213-219.
- [136] MARC DE NEYER; GOREZ, R. & BARRETO, J. M. "Disturbance Rejection Based on Fuzzy Models". In: *Decision Support Systems and Qualitative Reasoning*, p. 215-220, North-Holland: Elsevier Science Publishers B. V., March, 1991.
- [137] MARTINI, A.; EHRIG, H. & NUNES, D. "Elements of basic category theory". Relatório Técnico 96-5, University Berlin, March, 1996.
- [138] MARTINS, R. C. B. "Lógica de Primeira Ordem e Desenvolvimento de Softwares". Relatório técnico ccr-081, IBM Brasil - Centro Científico do Rio de Janeiro, Rio de Janeiro, Outubro, 1989.
- [139] MARTINS, R. C. B. & MOURA, A. *Desenvolvimento Sistemático de Sistemas Corretos: A Abordagem Denotacional*. Campinas : VI Escola de Computação, 1989.
- [140] MAULSBY, D. & WITTEN, I. H. "Teaching Agents to Learn: From User Study to Implementation". *Computer*, n. 30, p. 36-44, 1997.
- [141] MAYES, J. T. *Cognitive Tools: a suitable case for learning*. Heidelberg: Springer-Verlag, 1992.
- [142] MCCARTHY, J. "A proposal for the Dartmouth Summer Research Project on Artificial Intelligence - John McCarthy, August, 1955". URL: <http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html>. 1996.
- [143] MCCULLOCH, W. S. *Embodiments of Mind*. Cambridge, Massachussets : The MIT Press, 1989.

- [144] MCCULLOCH, W. S. & PITTS, W. H. "A logical calculus of ideas immanent in nervous activity". *Bull. of Mathematical Biophysics*, n. 5, p. 115-133, 1943.
- [145] MEAD, C. A. "silicon models of neural computation". In: *IEEE First International Conference on Neural Networks*, 1987, vol. 1, p. 93-106.
- [146] MEAD, C. A. *Analog VLSI and Neural Networks*. New York : Addison-Wesley, 1989.
- [147] MEAD, C. A. & MAHOWALD, M. A. "A silicon model of early visual processing". *Neural Networks*, n. 1, p. 234-278, 1988.
- [148] MEDLER, D. A. "A Brief History of Connectionism". URL: http://www.icsi.berkeley.edu/~jagota/NCS/VOL1/P3_html/vol1_3.html, 1999.
- [149] MENEZES, P. B. *Linguagens Formais e Autômatos*. Porto Alegre : Sagra Luzzato, 2000.
- [150] MIDDELBURG, C. A. "VVSL Specification of a Transaction-oriented Access Handler". In: *Specifications of Database Systems, Workshops in Computer Series*, Springer-Verlag, 1992.
- [151] MIDDELBURG, C. A. *Logic and Specification*. Computer Science: Research and Practice 1. Chapman and Hall, 1993.
- [152] MINSKY, M. L. & PAPERT, S. A. *Perceptrons: an introduction to computational geometry*. Massachussets : The MIT Press, 1969.
- [153] MINSKY, M. L. & PAPERT, S. A. *Perceptrons: an introduction to computational geometry*. Massachussets : The MIT Press, 3ª ed., 1988. (Impressão modificada do original de 1969).
- [154] MOUSSALLE, N. M.; VICCARI, R. M. & CORREA, M. *Intelligent Tutoring Systems modelled through the mental states*, cap. 3 : Advances in Artificial Intelligence, 1996.
- [155] NADEAU, D. R. "Building Virtual Worlds with VRML". In: *IEEE Computers Graphics and Applications*, J. J. Thomas, Ed., vol. 19, p. 18-29, Los Alamitos, CA: IEEE Computer Society, April, 1999.
- [156] NELSON, D. A. & ROSSITER, B. N. *Prototyping a Formal Object-Oriented Database in P/FDM*. Relatório técnico, Dept. of Computing Science, University of Newcastle upon Tyne, 1995.

- [157] NEWELL, A. & SIMON, H. "The Logic Theory Machine". *IRE Transactions on Information Theory*, vol. 2, n., p. 61-79, 1956.
- [158] NEWELL, A. & SIMON, H. "GPS a program that simulates human thought". In: *Computers and thought*, p. 279-293, Ney York: Academic Press, 1963.
- [159] NICOLACI-DA COSTA, A. M. *Na malha da Rede - Os impactos íntimos da Internet*. Rio de Janeiro : Editora Campus, 1998.
- [160] NIEVOLA, J. C. *Sistema Inteligente para Auxílio ao Ensino em Traumatologia Crânio-Encefálica*. Tese de Doutorado, Programa de Pós-graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina - UFSC, 1995.
- [161] NILSSON, N. J. "Artificial Intelligence: A New Synthesis". URL: <http://www.cs.utu.fi/knuutila/courses/ai/98/>, 1998.
- [162] PAGANO, R. & BARRETO, J. M. "Laboratory experiment simulation in electrical engineering education". In: *Proceedings of ESC'89 - 3rd European Simulation Congress*, Edinburgh, September, 1989, p. 311-317.
- [163] PAGANO, R. & BARRETO, J. M. "A theoretical model of hipertext". In: *Proceedings of AINN'90 - International Conference of Artificial Intelligence Applications and Neural Networks*, Zurich, 1990, Acta Press, M.H. Hanza Edition, p. 10-15.
- [164] PAGANO, R. & BARRETO, J. M. "Psychodrame pédagogique et hypermédia". In: *Proceedings of Sixième Congrès APTLF*, Brussels, May, 1990.
- [165] PAGANO, R. & BARRETO, J. M. "Hypermedia Supporting a Psychodramatic Experience: a case study". In: *Proceedings of 8th ICTE'91 - The Eighth International Conference on Technology and Education*, Toronto, May, 1991, vol. 2, p. 85-87.
- [166] PAGANO, R. & BARRETO, J. M. "Hypertext Information Technology in Medical Education". In: *Proceedings of 6th Mediterranean Electrotechnical Conference - IEEE-MELECON'91*, Yugoslavia, May, 1991, vol. 2, p. 1577-1580.
- [167] PAGANO, R. & BARRETO, J. M. "Computer Simulation + Information Technology: Good tools for classroom Work?". In: *9th International Conference on Technology and Education - ICTE'92*, Paris, 1992, vol. 3, p. 16-20.

- [168] PAGANO, R.; LEBACQ, J. & BARRETO, J. M. "Field testing SIMED: Evaluation of a Framework to Create Classroom Experiments in Physiology". *Archives Internationales de Physiologie et de Biochimie*, n. 100, p. 43, 1992.
- [169] PAGANO, R. L. *Computer Simulation as an educational tool*. Tese de Doutorado, Faculty of Applied Sciences, University of Louvain la Neuve, Belgium, 1992.
- [170] PAGANO, R. L. & BARRETO, J. M. "Hypertext and the Teaching process". Relatório técnico, Facultés Universitaires Notre Dame de la Paix - FUNDP, Instituto de Informática, FUNDP/INF/RP90/4, Namur, Bélgica, 1990.
- [171] PALINSCAR, A. S. & BROWN, A. L. "Reciprocal teaching of comprehension fostering and comprehension-monitoring activities". *Cognition and Instruction*, n. 1, p. 117-175, 1984.
- [172] PANTELIDIS, V. S. "Virtual Reality and Engineering Education". In: *Computer Applications in Engineering Education*, vol. 5, New York: Wiley-Interscience, 1997.
- [173] PAPERT, S. *A máquina das crianças: repensando a Escola na era da Informática*. Porto Alegre, RS : Artes Médicas, 1994.
- [174] PENROSE, R. *The Emperor's New Mind: Concerning, Computers, Minds, and Laws of Physics*. New York : Penguin Books, 1991.
- [175] PIAGET, J. *A construção do real na criança*. Rio de Janeiro : Zahar Editores, 1963.
- [176] PIAGET, J. *O julgamento moral na criança*. São Paulo : Editora Mestre Jou, 1977.
- [177] PIAGET, J. *The Equilibrium of Cognitive Structures: The Central Problem in Cognitive Development*. Chicago, Illinois : University of Chicago Press, 1985.
- [178] PILKINGTON, R. & GRIERSON, A. "Generating Explanations in a Simulation-Based Learning Environment". *International Journal of Human-Computer Studies*, n. 45, p. 527-551, 1996.
- [179] PILKINGTON, R. & PARKER-JONES, C. "Interacting With Computer-Based Simulation: The Role Of Dialogue in Computer Based Learning Unit". LS2 9JT, University of Leeds, Leeds, England, 1996.

- [180] PYLYSHYN, Z. *Computation and Cognition: Toward a Foundation for Cognitive Science*. Cambridge, MA : MIT Press, 1985.
- [181] PYLYSHYN, Z. W. *Computation and Cognition: Toward a Foundation for Cognitive Science*. Cambridge, Massachusetts : A Bradford Book, The MIT Press, 2 ed., 1989.
- [182] RAMIREZ POZO, A. T. *Um Sistema de Ensino Inteligente via Sociedade de Multi-agentes aplicado ao diagnóstico de Epilepsia*. Tese de Doutorado, Programa de Pós-graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina - UFSC, 1996.
- [183] RAMOS, E. M. F. *Análise ergonômica do sistema hiperNet buscando o aprendizado da cooperação e da autonomia*. Tese de Doutorado, Programa de Pós-graduação em Engenharia de Produção, Universidade Federal de Santa Catarina - UFSC, 1996.
- [184] RAPAPORT, W. J. "Cognitive Science". In: *Encyclopedia of Computer Science Third Edition*, Van Nostrand Reinhold, 1993.
- [185] RAUGHUNATHAN, S. "Qualitative Reasoning About Approximations in Quantitative Modeling". *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, n. 27, p. 683-690, September, 1997.
- [186] REZABEK, L. L. "Distance Education and Telecommunications technologies: lessons for educators". In: *The Ninth International Conference on Technology and Education*, March, 1992, N. Estes & M. Thomas, Eds., vol. 3.
- [187] RICH, E. & KNIGHT, K. *Inteligência Artificial*. São Paulo : Makron Books, 2ª ed., 1994.
- [188] ROSENBLATT, F. "The Perceptron: a probabilistic model for information storage and organization in the brain". *Psychological Review*, n. 65, p. 386-408, 1958.
- [189] ROSNAY, J. "History of Cybernetics and Systems Science". URL: <http://pespmc1.vub.ac.be/CYBSHIST.html>, 1996.
- [190] RUSSEL, S. J. & NORVIG, P. *Artificial Intelligence: a modern approach*. London, UK : Prentice Hall, 1ª ed., 1995.
- [191] SALLES, P. E. A. "Deriving Explanations from Qualitative Models". In: *Proceedings of the 8th World Conference on Artificial Intelligence in Education*, Japan, August, 1997, p. 474-481.

- [192] SANNELLA, D. & TARLECKI, A. "Essential Concepts of Algebraic Specification and Program Development". *Formal Aspects of Computing*, n. 9, p. 229-269, 1997.
- [193] SCARDAMALIA, M.; BEREITER, C. & STEINBACH, R. "Teachability of reflective processes in written composition". *Cognitive Science*, n. 8, p. 173-190, 1984.
- [194] SCOTT, D. *Relating theories of the lambda calculus. Essays on Combinatory Logic, Lambda Calculus and Formalisms*. Academic Press, 1980.
- [195] SHANNON, C. E. & MCCARTHY, J. *Automata Studies*. New Jersey : Princeton University Press, 1956.
- [196] SHAPIRO, S. Artificial intelligence. In: *Encyclopedia of Artificial Intelligence*, New York: John Willey & Sons, 1991.
- [197] SHERRY, L. & MORSE, R. "An Assessment of Training Needs in the Use of Distance Education for Instruction". *International Journal of Educational Telecommunications*, n. 1, p. 5-22, 1995.
- [198] SHORTLIFFE, E. H. *MYCIN: a rule-based computer program for advising physicians regarding antimicrobial therapy selection*. Tese de Doutorado, Stanford University, California, 1974.
- [199] SHORTLIFFE, E. H. *Computer based medical consultation: MYCIN*. New York : American Elsevier, 1976.
- [200] SIMON, H. A. *Why Would Machines Learn?*, cap. 2, p. 25-36. New York: Springer-Verlag, 1984.
- [201] SINGH, M. P. & HUHNS, M. N. "Internet-based Agents: applications and infrastructure". *The Internet Computing*, n. 4, p. 8-20, August, 1997.
- [202] SKINNER, B. F. *Science and human behavior*. New York : The Macmillan Company, 1953.
- [203] SKINNER, B. F. "Teaching Machines". *Science*, n. 128, p. 969-977, 1958.
- [204] SOWA, J. F. *Conceptual Structures: Information Processing in Mind and Machine*. New York, USA : Addison-Wesley Publishing Company, Inc., 1984.
- [205] SPENCER, D. "A Survey of Categorical Computation: Fixed Points, Partiality, Combinators ... Control?". *Bulletin of the European Association for Theoretical Computer Science*, n. 43, p. 285-312, 1991.

- [206] STEVENS, A. & COLLINS, A. "The goal structure of a socratic tutor". Relatório Técnico 3518, Bold Beranek and Newman, Cambridge, Massachusetts, 1977.
- [207] SUPPES, P.; JERMAN, M. & BRIAN, D. *Computer-Assisted Instruction: Stanford's 1965-66 Arithmetic Program*. New York : Academic Press, 1968.
- [208] TANENBAUM, A. S. *Redes de Computadores*. Editora Campus, 2^a. ed., 1994.
- [209] TARBY, J. C. "The Human-Computer Dialogue in Learning Environments". *Computer Applications in Engineering Education*, n. 5, p. 29-39, February, 1997.
- [210] TOYNBEE, A. *A Sociedade do Futuro*. Rio de Janeiro : Zahar Editores, 1974.
- [211] TRNKOVÁ, V. "Automata and Categories". In: *Mathematical Foundations of Compute Science*, Lecture Notes in Computer Science, p. 138-152, Springer-Verlag, 1975.
- [212] TWIGGER, D. E. A. "The conceptual change in science project". *Journal of Computer Assisted Learning*, n. 7, p. 144-155, 1991.
- [213] VANGHELUWE, H.; BARRETO, J. M. & VANSTEENKISTE, G. C. *Application of a Multifaceted Modelling Methodology: An Example in Physiology*, p. 233-238 : IMACS, 1991.
- [214] VEDANA, S. B. *Ambiente Computacional para Ensino de Neurofisiologia*. Trabalho de conclusão de Curso de Bacharelado em Informática, 2001. (Universidade do Planalto Catarinense, Lages, SC).
- [215] VETTER, R. & SEVERANCE, C. "Web-based Education Experiences". *IEEE-Computer*, n. 30, p. 139-143, November, 1997. (Edição especial: The Changing Face of Education).
- [216] VON STAA, A. *Engenharia de Programas*. Rio de Janeiro : LTC - Livros Técnicos Científicos Editora S. A., 1983.
- [217] WANG, H. "Intelligent Agents in the Nuclear Industry". *IEEE Computer*, n. 11, p. 28-44, November, 1997.
- [218] WATERS, C. "The programmer's apprentice project: A research overview". Relatório Técnico 1004, Massachusetts Institute of Technology - Artificial Intelligence Laboratory, Massachusetts, USA, November, 1987. AI-MEMO.

- [219] WIENER, N. *Cibernética e Sociedade*. Ed. Cultrix, 1954.
- [220] WILSON, J. R. "Como e o que Aprendemos?". In: *A Mente*, p. 105-113, Rio de Janeiro: Livraria José Olympio Editora, 1976.
- [221] WILSON, J. R. "Máquinas para enfrentar a crise". In: *A Mente*, p. 122-123, Rio de Janeiro: Livraria José Olympio Editora, 1976.
- [222] WINOGRAD, T. "What does it mean to understand language?". In: *Perspectives on Cognitive Science*, London: Ablex Publishing, 1981.
- [223] WINSTON, P. *Inteligência Artificial*. São Paulo : LTC, 1981.

Índice Remissivo

- alfabeto, 65
- alfabeto de entrada, 89
- alfabeto de saída, 90
- algoritmos de aprendizado, 104
- ambientes de Ensino-aprendizagem, 4
- Análise de Oportunidades, 113
- Análise Funcional, 113
- aprendizado animal, 34
- aprendizado de máquina, 34, 42
- aprendizado evolutivo, 54
- aprendizado humano, 41
- Aprendizado Não Supervisionado, 51
- aprendizado programado, 11
- Aprendizado Supervisionado, 51
- autômato, 87
- autômato finito, 87
- Auto-associador, 51
- Avatar, 28

- Behaviorismo, 11, 35
- BUGGY, 25

- Categoria de Estados de um Hipertômato, 94
- Categoria dos Autômatos, 60, 94
- Categoria dos Conceitos, 60, 77
- Categoria dos Conjuntos, 60
- Categoria dos Estados de um Autômato, 94
- Categoria dos Hipertômatos, 94, 95
- Ciência Cognitiva, 34
- Ciência Cognitiva Computacional Conexionista, 40

- Ciência da Computação, 38, 86
- Ciberespaço, 28
- Ciclo de Vida, 2, 6, 32
- classificação do aprendizado, 50
- cognição, 34
- Cognição e Computação, 39
- Cognitivismo, 35
- Cognitivismo Simbólico Computacional, 39
- compilação, 68
- compiladores, 68
- composição associativa de funtores, 74
- composição de funtores, 74
- Computação Evolucionária, 30
- computabilidade, 58
- conhecimento dedutivo, 52
- conhecimento fatorial, 52, 53
- conhecimento hábil, 52
- conhecimento superficial, 53
- conjunto de nós, 89
- construção de hipertômato, 101

- diagrama, 72
- DIDAK, 15
- Dualidade, 74

- EDUCOM, 17
- Engenharia de Programas, 5, 6, 27, 32, 63, 64, 74, 98, 112, 140, 141
- Engenharia de Programas Educacionais, 33, 141
- Engenharia de Programas Educacionais Hiperfídia, 6

- Engenharia de Programas em IA, 32
Ensino com computador, 1, 3, 6
Ensino programado, 16
Ensino-aprendizagem, 8, 18, 31, 32, 52
epimórficos, 74
epimorfismo, 73
estado, 89
estado cognitivo, 80
estado inicial, 89
estado observável, 91
Estados cognitivos, 79
estratégias de aprendizado, 43
Exemplos de alfabetos, 65
- Filosofia Computacional, 38
função computável, 56, 57
função de transição, 89
função de transição de estados, 84
funções, 76
funções injetoras, 76
funções sobrejetoras, 76
funtor, 74
funtor esquecedor, 75
funtor identidade, 74
- grafo, 88
grafo orientado, 87
Gramáticas Gerativas, 68
GUIDON, 25
- Hetero-autoassociador, 51
Hipermissão, 14, 30, 33
hipermissão como autômato, 90
Hipertômatos, 94, 95
hipertexto, 89
- IBM, 20
ICEM, 4
Informática na Educação, 17
INTEGRATION, 24
- Inteligência Artificial Distribuída, 53
Inteligência Artificial no Ensino, 24
Inteligência Computacional no Ensino
 com Multimídia, 4
Inteligência Evolutiva, 53
INTERNET, 30
INTERNET 2, 30
isomórficos, 74
- KADS, 32
- Linguagens e Gramáticas Formais, 68
linguagens formais, 68
LOGO, 26
- Máquina de Crowder, 14
máquina de estados finitos, 86
Máquina de Pask, 14
Máquina de Pressey, 10
Máquina de Turing, 40
Máquinas de Skinner, 11
método de Jackson, 5
métodos formais, xv, 62
MENO-TUTOR, 25
Modelagem Cognitiva, 50
monomórficos, 74
monomorfismo, 73
morfismos, 71
MUMATH, 25
MYCIN, 24
- Nós essenciais de um Hipertômato, 99
objetos, 67
opinário, 113
- Paradigmas da Ciência Cognitiva, 39
Percurso Composto Associativo, 94
Percurso Identidade, 94
Piaget, 16, 26, 81
PLATO, 21

- Problema da Parada, 58
PROFCOMP, 25
programação ramificada, 14
Projeto ICEM, 101
Psicologia Cognitiva, 38
Psicologia Computacional, 37
- Redes Neurais, 2, 35
Regra identidade, 68
representação do conhecimento, 47
RNA, 104
- SCHOLAR, 24
SIMED, 25
simulação, 25
simuladores, 29
Sistema Dinâmico, 84
Sistema Dinâmico Contínuo no Tempo, 85
Sistema Dinâmico Discreto no Tempo, 85
sistema estacionário, 86
sistema funcional temporal, 84
Sistema invariante no tempo, 85
sistema invariante no tempo, 86
sistema simbólico, 39
sistemas de Ensino-aprendizagem, 4
Sistemas Especialistas, 24, 50
Sistemas Formais, 60
sistemas hipermídia, 4
SMALLTALK, 26
SOCRATES, 20
SOPHIE, 24
- Taxinomias, 49
Teoria Construtivista, 16
Teoria da Computação, 68
Teoria das Categorias, 5, 33, 60, 62-64, 70, 76, 81, 82
Teoria dos Autômatos, 68
- Tipos de morfismos, 72
transição de estados, 87
Turing, 35
TUTOR, 21
WHY, 25