

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE
ELETRICIDADE

ASSISTENTE DE RESOLUÇÃO DE
PROBLEMAS PARA O SISTEMA
MATHNET

Hélder Pereira Borges

São Luis

2002

ASSISTENTE DE RESOLUÇÃO DE PROBLEMAS PARA O SISTEMA MATHNET

Dissertação de Mestrado submetida à Coordenação do programa de Pós-Graduação
em Engenharia de Eletricidade da UFMA como parte dos requisitos para
obtenção do título de mestre em Ciência da Computação.

Por

Hélder Pereira Borges

Maio, 2002

ASSISTENTE DE RESOLUÇÃO DE PROBLEMAS PARA O SISTEMA MATHNET

Hélder Pereira Borges

DISSERTAÇÃO APROVADA EM 03 / 05 / 2002

Prof. Dr. Sofiane Labidi

(Orientador)

Prof. Dr. Roberto da Silva Bigonha

(Membro da Banca Examinadora)

Prof. Dr. Zair Abdelouahab

(Membro da Banca Examinadora)

ASSISTENTE DE RESOLUÇÃO DE PROBLEMAS PARA O SISTEMA MATHNET

MESTRADO

Área de Concentração: CIÊNCIA DA COMPUTAÇÃO

Hélder Pereira Borges

Orientador: Dr. Sofiane Labidi

Programa de Pós-Graduação
Em Engenharia de Eletricidade da
Universidade Federal do Maranhão

DEDICATÓRIA

A Deus Pai,

A Deus Filho,

A Deus Espírito Santo,

À minha linda esposa,

E a meus pais.

AGRADECIMENTOS

A Deus, o Todo poderoso dono da minha vida.

A Débora, por suportar sem vacilar, todos os momentos de desânimo.

Aos meus pais, Dalmo e Maria Dulce, fundamentais em toda minha formação.

A minha Vó Deusina, por sempre querer o melhor pra mim.

Aos meus sogros, Guaribaldi e Nerilva, pelo incentivo constante.

Aos meus irmãos, cunhado e cunhadas por me aturarem.

Ao Prof. Dr. Sofiane Labidi, pela orientação e confiança independente de circunstâncias e tempo.

A Luciano Coutinho, pelo tempo dispensado e as inúmeras elucidações.

À Coordenadoria de Pós-Graduação em engenharia de eletricidade, pelo auxílio no decorrer dos anos.

A todos aqueles que, direta ou indiretamente, colaboraram e acreditaram na realização deste trabalho.

RESUMO

Nos dias atuais, o ensino à distância computadorizado tem crescido de forma generalizada. Este crescimento é decorrente dos avanços tecnológicos obtidos no âmbito de redes de computadores, com destaque para Internet.

O Sistema MATHNET, resultado da integração dos paradigmas da Aprendizagem Assistida por Computador e Aprendizagem Cooperativa, integração também conhecida como CSCL (Computer Supported Cooperative Learning), implementa um modelo computacional para ambientes interativos de Ensino e Aprendizagem Cooperativa com base em múltiplos agentes artificiais e humanos, dispostos em uma estrutura de rede de computadores, utilizando diversos recursos multimídia.

Um componente importante para os sistemas de ensino é o Assistente de Resolução de Problemas que tem como objetivo a fixação de conteúdo e avaliação no processo de ensino e aprendizagem.

Um Assistente de Resolução de Problemas tem como principais objetivos, armazenar e resolver problemas propostos, dar sugestões aos aprendizes, quando solicitado e ainda verificar as etapas de uma dada solução de problema quanto sua veracidade.

Apresentando as funcionalidades de Resolver Problemas e Analisar Soluções, o Assistente de Resolução de Problemas é uma ferramenta de grande importância no processo de Ensino e Aprendizagem, considerando-se a necessidade de fixação e avaliação do grau de aprendizagem dos conteúdos ministrados.

Neste trabalho, tendo em vista a necessidade do Sistema MATHNET de uma ferramenta que auxilie na avaliação do grau de aprendizagem dos seus alunos e grupos, que auxilie na fixação do conteúdo e também tire dúvidas e sugira procedimentos, apresenta-se um Assistente de Resolução de Problemas, que se propõe a desempenhar estas funções.

Palavras-Chave: Sistemas Tutores Inteligentes, Ensino e Aprendizagem, Resolução de Problemas, MATHNET.

ABSTRACT

Nowadays, the distant computerized teaching has grown worldwide. Such growth is due to technological advances obtained on computer network scopes, mainly on the Internet.

The MATHNET System, resulting from the integration of the Computer Assisted Learning and Cooperative Learning paradigms, integration also known as CSCL (Computer Supported Cooperative Learning), implements a computer model to interactive environment of Cooperative Teaching and Learning based on multiple artificial and human agents, placed on a computer network structure, making use of several multimedia resources.

An important component to Teaching Systems, the Solving Problems Assistant, has as goal, the fixation of subject and evaluation in the Teaching and Learning process.

A Solving Problems Assistant has as its main purposes to store, solve proposed problems, make suggestions to the apprentices when required and even verify the steps of a problem solution regarding its veracity.

Showing the functionalities of Solving Problems and Analyzing Solutions, the Solving Problem Assistant is a tool of great importance on the Teaching and Learning Process, regarding the fixation and evaluation needs of the learning level of the subject taught.

This Msc thesis, considering the MATHNET System's need of a tool that could help on the evaluation of the students and groups learning level, helping on the subject learning and also answering questions and suggesting procedures, a Solving Problems Assistant is presented so as to fulfill those requirements.

Key-words: Intelligent Tutoring Systems, Teaching and Learning, Problems Solve, MATHNET.

SUMÁRIO

LISTA DE FIGURAS	vi
LISTA DE TABELAS	ix
1. INTRODUÇÃO.....	1
1.1 Contexto	2
1.2 Objetivo.....	3
1.3 Justificativa e Relevância	4
1.4 Organização da Dissertação.....	5
2. MATHNET.....	6
2.1 Ambiente Mathnet.....	6
2.2 Arquitetura Multiagentes.....	8
2.2.1 Conceito de Agente	9
2.2.2 Agentes MATHNET	11
2.2.3 Núcleo Padrão de Agentes Artificiais MATHNET	14
2.3 Interatividade	15
2.4 Atividades Pedagógicas	17
2.5 Estratégias Pedagógicas	20
2.5.1 Estratégia Global	21
2.5.2 Estratégias Específicas de Atividades	22
2.5.3 Critérios de seleção das estratégias pedagógicas	23
2.7 O processo de avaliação.....	25
2.8 Considerações Finais	27
3. RESOLUÇÃO DE PROBLEMAS.....	28

3.1 O que são problemas	29
3.2 Tipos de problemas.....	30
3.3 Métodos para resolução de problemas	34
3.3.1 Método de George Pólya.....	35
3.3.2 Método do Serway.....	45
3.3.3 Mapa Mental.....	48
3.4 Considerações Finais	55
4. RESOLUÇÃO DE PROBLEMAS NO MATHNET	56
4.1 Atividades Pedagógicas	56
4.2 Metodologia	58
4.3 Arquitetura Mathnet.....	61
4.4 Modelagem do Processo de Resolução de Problemas	63
4.4.1 Caso de Uso Professor.....	63
4.4.2 Caso de Uso Aprendiz.....	64
4.5 Caso de Uso Resolver Problema	65
4.5.1 Fluxo Principal da Resolução de Problemas.....	65
4.5.2 Fluxos Alternativos da Resolução de Problemas	66
4.6 Considerações Finais	68
5. ASSISTENTE DE RESOLUÇÃO DE PROBLEMAS.....	69
5.1 Análise de Requisitos.....	69
5.1.1 Determinação do Contexto	71
5.1.2 Determinação do Escopo	72
5.1.3 Definição dos Requisitos	73
5.1.4 Detalhamento dos Requisitos de Interface.....	77
5.1.5 Detalhamento dos Requisitos Não-Funcionais.....	83

5.1.6 Classificação dos Requisitos	83
5.1.7 Revisão dos Requisitos.....	84
5.2 Estrutura do Assistente Resolução de Problemas.....	85
5.2.1 Classe Problema.....	86
5.2.2 Classe Solução	87
5.2.3 Diagrama de Seqüência	87
5.3 Processo de Resolução de Problemas.....	89
5.3.1 Aluno Resolvendo Problemas.....	89
5.3.2 Assistente Resolvendo Problemas.....	91
5.3.3 Assistente Analisando Solução	91
5.4 Considerações Finais	92
6. RESULTADOS OBTIDOS.....	93
6.1 Resolução de Problemas	94
6.2 Verificação de Soluções	97
6.3 Considerações Finais	99
7. CONCLUSÃO	100
BIBLIOGRAFIA.....	103

LISTA DE FIGURAS

Figura 2.1 - Ambiente MATHNET	7
Figura 2.2 - Arquitetura Multiagentes MATHNET	11
Figura 2.3 - Interface principal do ZEUS Agent Toolkit.....	15
Figura 2.4 - Atividades Pedagógicas.....	18
Figura 3.1 – Tipos de Problemas (CommonKads).....	31
Figura 3.2 – Processo de Design	31
Figura 3.3 – Processo de Planejamento	32
Figura 3.4 – Processo de Atribuição.....	33
Figura 3.5 – Processo de Diagnose.....	33
Figura 3.6– Processo de Avaliação	34
Figura 3.7 – Método do Pólya.....	35
Figura 3.8 – Equação simples	41
Figura 3.9 – Equação substituída	42
Figura 3.10 – Equação 3	43
Figura 3.11 – Exemplo de Resolução de Problema pelo método do Pólya.....	45
Figura 3.12 - Método de Serway.....	47
Figura 3.13 – Exemplo de Resolução de Problema pelo método do Serway.....	48
Figura 3.14 - Anotação 1.....	49
Figura 3.15 - Anotação 2.....	49
Figura 3.16 – Mapa Conceitual Aranha.....	51
Figura 3.17 – Mapa Conceitual Hierárquico	51
Figura 3.18– Mapa Conceitual FlowChart	51
Figura 3.19 – Mapa Conceitual Aranha.....	52
Figura 3.20 – Exemplo Mapa Mental.....	53

Figura 3.21 – Mapa Mental Problema	54
Figura 4.1 – Atividades pedagógicas que podem utilizar o Assistente	57
Figura 4.2 – Relacionamento entre o método de Pólya e do Serway.....	58
Figura 4.3 – Metodologia de Resolução de Problemas do Mathnet.	61
Figura 4.4 – Proposta Arquitetura Mathnet.....	62
Figura 4.5 – Diagrama do caso de uso Professor	63
Figura 4.6 – Diagrama do Caso de uso Aprendiz	64
Figura 5.1 – Atividades da Análise de Requisitos	69
Figura 5.2 – Contexto do Assistente	71
Figura 5.3 – Casos de Uso.....	74
Figura 5.4 – Atores.....	75
Figura 5.5 – Relacionamento Aprendiz – Lista de problemas.....	76
Figura 5.6 - Relacionamento Professor – Cadastro de Problemas.....	76
Figura 5.7 - Relacionamentos ARP	76
Figura 5.8 – Diagrama de Contexto	77
Figura 5.9 – Interface de Cadastro de Problemas	78
Figura 5.10 – Interface da Criação / Carga da Lista de Problemas.....	79
Figura 5.11 – Interface da Lista de Problemas	79
Figura 5.12– Interface Resolver Problemas	80
Figura 5.13 – Interface Especificar Objetivo	81
Figura 5.14 – Interface Especificar Dados	81
Figura 5.15 – Interface Especificar Conhecimento.....	82
Figura 5.16 – Interface elaborar Estratégia.....	82
Figura 5.17 – Visão Geral das Classes que compõe o ARP	85
Figura 5.18 – Classe Problema	86
Figura 5.19 – Classe Solução	87
Figura 5.20 – Diagrama de Seqüência da Resolução de Problemas.....	88

Figura 5.21 – Aluno Resolvendo Problema.....	89
Figura 5.22 – Aluno Resolvendo Problema.....	90
Figura 5.23 – Assistente Resolvendo Problema	91
Figura 5.24 – Assistente Analisando uma Solução.....	92
Figura 6.1 – Exemplo Resolução Problema 1.....	94
Figura 6.2 – Resolução Problema 2	95
Figura 6.3 – Exemplo Resolução Problema 3.....	96
Figura 6.4 – Solução Correta do Problema 1.....	97
Figura 6.5 – Solução Correta do Problema 2.....	98
Figura 6.6 – Solução Errada do Problema 1	98
Figura 6.6 – Solução Errada do Problema 2.....	99

LISTA DE TABELAS

Tabela 5.1 - Missão	72
Tabela 5.2 - Limites.....	72
Tabela 5.3 - Benefícios	73
Tabela 5.4 – Casos de uso	74
Tabela 5.5 - Atores	75
Tabela 5.6 – Requisitos para Interface do Assistente	77
Tabela 5.7 – Requisito de Desempenho	83
Tabela 5.8 – Requisito de Desenho	83
Tabela 5.9 – Atributo de Qualidade (Segurança)	83
Tabela 5.11 – Alguns Requisitos Classificados.....	84

1. INTRODUÇÃO

Graças aos avanços tecnológicos alcançados nas últimas décadas, caminhamos para formas de gestão menos centralizadas, mais flexíveis, integradas, onde haverá menos pessoas, porém, trabalhando mais sinergicamente em estruturas enxutas.

A grande tendência atual nos sistemas educacionais (conforme Moran, 2000) é uma reorganização física dos prédios, onde provavelmente, haverá uma menor quantidade de salas de aula, porém mais funcionais, por possuírem acesso à internet e intranets.

A utilização dos computadores pelos alunos para pesquisa e busca de novos materiais para solução de problemas se tornam cada vez maior e, continuamente, crescem em número de adeptos.

Os professores, parte fundamental do processo, também têm sua contribuição, estando conectados à internet, em casa, na sala e também com recursos tecnológicos para exibição de materiais de apoio para motivar os alunos e ilustrar as suas idéias.

O fruto desse avanço de concepção é que teremos mais ambientes de pesquisa grupal e individual em cada escola; as bibliotecas se convertem em espaços de integração de mídias, software e bancos de dados.

Com todos estes acontecimentos, espera-se uma integração profunda entre a sociedade e a escola, entre a aprendizagem e a vida. A aula não será um espaço inanimado, mas tempo e espaço contínuos de aprendizagem.

Uma grande vantagem, e de valor inestimável, é a flexibilidade, pois os cursos terão características híbridas em relação a estilo, presença, tecnologias e requisitos. Existirão matérias somente presenciais; outras, apenas predominantemente e ainda aquelas predominantemente virtuais e até mesmo unicamente.

O importante será aprender e não impor um padrão único de ensino.

O Ensino Computadorizado a Distância toma partido do aumento da velocidade e de largura de banda, que torna possível ver e ouvir a distância, viabilizando desta forma sua proposta.

As aulas poderão ser ministradas pelo professor de sua sala que será visto pelos alunos onde eles estiverem, podendo haver interação destes com o professor ou outros alunos. As aulas ficam disponíveis em locais de acesso público para consultas posteriores quando necessário for.

Em razão do avanço do ensino a distância e do ensino computadorizado, faz-se necessário um mecanismo para resolução de problemas, para servir como meio de fixação da aprendizagem e ajudar na avaliação da mesma.

Uma Ferramenta de Resolução de Problemas pode ser considerada indispensável para um Sistema de Ensino Computadorizado, à distância ou não, devido os processos automatizados de resolver problemas e analisar soluções que são oferecidos em um mecanismo deste tipo. Um outro aspecto que também o torna necessário é a possibilidade de usar suas habilidades para outros fins, como por exemplo, para auxiliar no processo de avaliação dos aprendizes.

1.1 Contexto

Este trabalho situa-se na área de Sistemas Tutores Inteligentes Cooperativos e faz parte do Projeto MATHNET. Este projeto é parte do programa PROTEM-CC em Informática na Educação, formado por um consórcio de três universidades: Universidade Federal do Maranhão – UFMA, Universidade Federal de Alagoas – UFAL e Universidade Federal de Santa Catarina – UFSC. Ele é mantido pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

O Projeto MATHNET tem por objetivo utilizar conceitos de Inteligência Artificial na educação a fim de desenvolver um Sistema Tutor Inteligente Cooperativo Computadorizado,

onde o processo de ensino-aprendizagem é realizado pela interação entre grupos de alunos (presenciais ou virtuais), professor e sistema.

Este trabalho trata especificamente de problemas propostos e suas soluções, visando à fixação e avaliação no processo de ensino-aprendizagem. Portanto, ele diz respeito aos processos do Assistente de Resolução de Problemas do sistema MATHNET, que é a ferramenta responsável por resolver problemas, inclusive, emitir ajudar ao aluno quando necessário e analisar soluções produzidas pelos aprendizes.

1.2 Objetivo

O objetivo deste trabalho é implementar um Assistente de Resolução de Problemas, a qual será utilizado no sistema MATHNET (Labidi et al., 2000a), com o intuito de auxiliar o processo de avaliar a aprendizagem do conteúdo ministrado e também auxiliar em pontos duvidosos para o aluno, durante a resolução de um problema proposto.

A partir da formulação de um problema pelo professor e especificação dos dados necessários para sua resolução, definiremos uma modelagem para eles, a partir da qual o Assistente será capaz de solucionar estes problemas.

Estabeleceremos, em nosso trabalho, direções gerais para a criação do Assistente de Resolução de Problemas e, a partir disso, definiremos os elementos arquiteturais de um Assistente de Resolução de Problemas específico.

Especificamente, pretende-se:

1. Definir uma modelagem para os problemas;
2. Definir uma modelagem para as soluções;
3. Especificar as funcionalidades da representação e análise de um problema e de uma solução;

4. Projetar e implementar o Assistente de Resolução de Problemas do MATHNET, fazendo uso da linguagem de programação Java (Walnum, 1997).

1.3 Justificativa e Relevância

A utilização da resolução de problemas nas escolas, para fixação dos assuntos das ministrados em aula e avaliação das disciplinas, é bastante utilizada desde os primórdios do ensino. Acredita-se que esta característica implicará em uma boa aceitação, em virtude de não ser algo novo, mas sim uma prática costumeira da maioria dos alunos.

As técnicas de resolução de problemas são bastante utilizadas e suas formas de solução devem ser adequadas ao aprendiz para um melhor aproveitamento, bem como para o sistema, visando a que este auxilie o aprendiz em suas dificuldades.

No Sistema MATHNET, é necessário um mecanismo que auxilie os processos de fixação e de avaliação dos resultados, em relação ao aprendizado dos conteúdos ministrados, visando ao aperfeiçoamento do processo de ensino e aprendizagem e, principalmente, objetivando a maximização do aproveitamento por parte do aprendiz. Este módulo estava previsto no Sistema MATHNET, mas somente agora ocorrerá sua efetiva implementação.

Existem poucas abordagens específicas sobre resolução de problemas acadêmicos por meio de sistemas computacionais, fato este responsável por grande impedimento ao desenvolvimento da pesquisa.

Desta maneira, nossa contribuição será de grande valia, acreditando-se que novas publicações serão acrescidas à produção científica existente sobre o tema resolução de problemas por meio computacional e ainda contribuindo com um exemplo efetivo de um mecanismo que cumpre os requisitos de resolver problemas e analisar soluções.

1.4 Organização da Dissertação

Esta dissertação está organizada em 7 capítulos. No capítulo a seguir, apresenta-se o Ambiente MATHNET de Ensino Inteligente Cooperativo Computadorizado, destacando-se sua arquitetura multi-agentes, a interatividade entre os componentes do sistema, as atividades pedagógicas realizadas, as estratégias pedagógicas que podem ser adotadas e ainda o processo de avaliação dos aprendizes.

No Capítulo 3, são abordadas definições a respeito do que são problemas e também uma revisão de algumas abordagens sobre técnicas de resolução de problemas. Estas técnicas são utilizadas como subsídios no processo de concepção, análise e implementação do Assistente de Resolução de Problemas.

As especificações de como todo o processo de resolução de problemas ocorre dentro do sistema MATHNET são feitas no Capítulo 4, destacando-se a metodologia adotada, arquitetura e casos de uso.

O Capítulo 5 foi destinado para as especificações do Assistente de Resolução de Problemas, levando-se em consideração os requisitos necessários, a estrutura do mesmo e os processos de resolver e analisar os problemas.

Os resultados obtidos através de teste, tanto em relação à resolução de problemas, quanto à análise de soluções, serão descritos no Capítulo 6.

Finalmente, no Capítulo 7, apresentam-se as conclusões e perspectivas futuras deste trabalho.

2. MATHNET

O projeto MATHNET é uma contribuição na área de ensino e aprendizagem cooperativa suportada por computador, possuindo como objetivo inicial avançar na direção da concepção e realização de um modelo computacional para ambientes interativos de Ensino Aprendizagem Cooperativa com base em múltiplos agentes artificiais e humanos, dispostos em uma estrutura de rede de computadores.

Pretendendo-se com isso integrar alunos, professores e sistema computacional em um espaço que sirva para promover o desenvolvimento de atividades cooperativas. Oferecendo, portanto, desta maneira, alternativas aos métodos pedagógicos tradicionais, beneficiando assim o aluno no processo de aquisição do conhecimento.

O nosso trabalho é proposto para o Ambiente Mathnet, ou seja, aos módulos já existentes no Mathnet, será acrescentada uma nova ferramenta já prevista, mas ainda não implementada.

Neste capítulo é destacada a maneira como é aplicada a metodologia de ensino cooperativo no Mathnet, a sua estrutura e seus principais conceitos, funcionalidades e objetivos, além de fornecer uma visão clara desse sistema e a aplicação de agentes artificiais, em especial na ferramenta de autoria.

2.1 Ambiente Mathnet

O ambiente MATHNET foi proposto através da junção de um Sistema Tutor Inteligente Multiagentes (Costa, 1997) com um Sistema Hiperímia de Ensino / Aprendizagem Cooperativo Computadorizado (Labidi et al., 1998).

O propósito do projeto é definir, projetar e implementar um ambiente computadorizado para um efetivo processo de aprendizagem cooperativa.

A etapa de implementação, que é a parte final do projeto e das diversas funcionalidades e estruturas do ambiente, está sendo finalizada atualmente, crendo-se ter logrado bom êxito com os resultados obtidos até então nas etapas já finalizadas, tendo-se em vista melhorias e acréscimos na qualidade do produto final.

O Ambiente MATHNET une as idéias que fundamentam os Sistemas Tutores Inteligentes aos recursos de uma rede de computadores, bem como ao paradigma de aprendizagem cooperativa, definindo um Ambiente de Ensino Inteligente Cooperativo Computadorizado, conforme ilustrado na figura 2.1.

O núcleo do Ambiente MATHNET encontra-se instalado no servidor que, operacionalmente, poderá estar distribuído pela rede. Aos serviços disponíveis no servidor, chamamos de Sistema MATHNET.

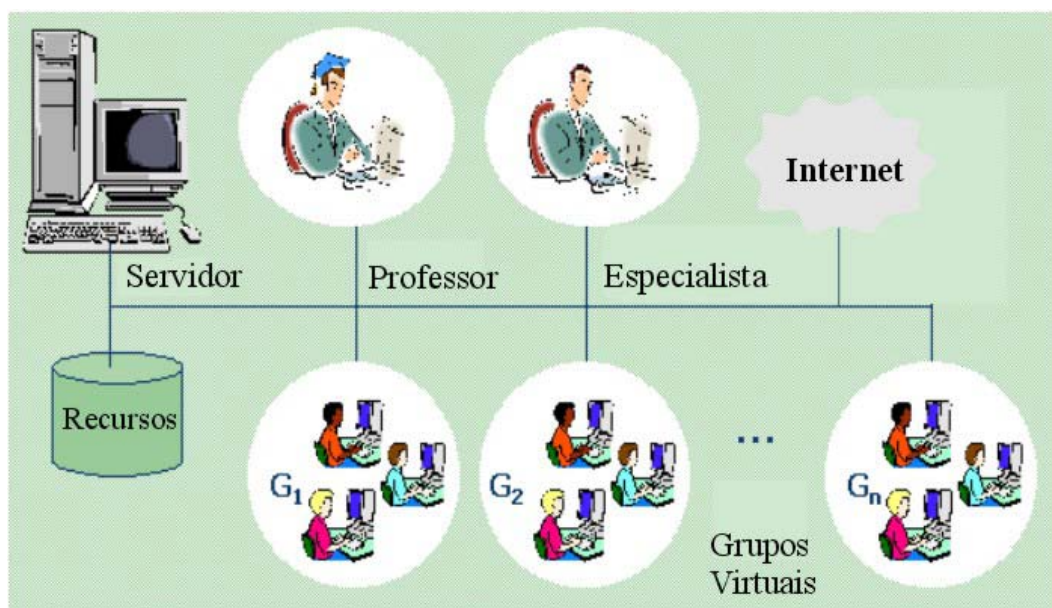


Figura 2.1 - Ambiente MATHNET

Os alunos são divididos em grupos distintos no ambiente MATHNET que são chamados de áreas cooperativas. Basicamente, ocorrem dois tipos de interações: a interação intragrupo, onde os estudantes cooperam e aprendem interagindo dentro de seus próprios grupos com os agentes artificiais e com o professor, e a interação intergrupo, que pode ocorrer entre os outros alunos nos diversos grupos existentes. Estas interações se dão através da utilização de recursos multimídia e da tecnologia de redes, intranets e internet .

O MATHNET tem a vantagem de prover uma cooperação em tempo real entre os alunos, usando mecanismos como o *chat* e a *vídeo conferência*. A interação também pode ser realizada de outras formas, através do correio eletrônico ou do *banco de dúvidas*, representando um *blackboard* compartilhado por todos, onde os alunos apresentam e respondem suas dúvidas.

Um grupo ou área cooperativa é formado normalmente por três alunos cooperando e interagindo com o ambiente (Labidi et al., 1998b), cada um a partir de um terminal. Desta maneira, os aprendizes podem realizar atividades sugeridas pelo professor, resolver problemas, solicitar informações ou tirar dúvidas, entre outras coisas.

Um professor interagindo com um terminal é considerado uma área cooperativa específica. Com o auxílio de recursos computacionais, o professor poderá, entre outras atividades, formar n grupos de alunos com base em alguns critérios pedagógicos e informações colhidas dos alunos, por exemplo, através de formulários e ainda determinar parâmetros básicos para o ensino do conteúdo (seqüência de atividades, tempo alocado para cada atividade) e outros parâmetros que auxiliarão o sistema no processo de avaliação dos aprendizes.

Em uma Sessão de Aprendizagem ocorre a interação entre várias áreas cooperativas interligadas através da rede, objetivando o aprendizado por parte dos alunos. Nas etapas da aprendizagem, existem diversos tipos de atividades que são desempenhadas pelos alunos e grupos.

2.2 Arquitetura Multiagentes

O Ambiente MATHNET é baseado em uma arquitetura Multiagentes composta por agentes inteligentes artificiais (Agente Tutor, Agente Estratégico, Agente de Modelagem do Aprendiz, Agentes de Domínio, etc) e por agentes humanos (aluno, grupos, professor e engenheiro do conhecimento) (Labidi et al., 2000b).

2.2.1 Conceito de Agente

Um agente pode ser definido como uma entidade física ou abstrata que executa uma ação sobre algo, seja sobre si ou sobre seu ambiente, produzindo um efeito (Ferber, 1988). Por se tratar de uma definição genérica, é necessário definir o contexto particular de aplicação da definição, para que seja possível especificar o conceito de agente adotado pelo MATHNET. É importante, então, deixar claro que o conceito adotado refere-se ao universo da Ciência da Computação.

No âmbito da Ciência da Computação pode-se falar de agentes artificiais como sendo sistemas de computação que executam determinadas tarefas com características que os enquadram como agentes. Agentes humanos são usuários humanos que projetam ou manipulam computadores (hardware ou software).

“Um agente é algo que percebe um ambiente através de sensores e age sobre esse ambiente através de efetadores” (Russell et al., 1995).

Russell e Norving (Russell et al., 1995) afirmam ainda que o trabalho de concepção de um agente inteligente consiste em definir o que será percebido pelo agente no ambiente, suas possíveis ações e, principalmente, os mecanismos através dos quais ele avaliará suas percepções para "escolher" a ação a ser executada, alcançando diferentes graus de autonomia.

Em (Wooldridge et al., 1995), os autores acrescentaram ao conceito de agentes inteligentes a noção de sociedade, ao afirmar que os agentes inteligentes devem ser capazes de se comunicar, interagindo uns com os outros através de algum tipo de linguagem comum.

Assim, avaliando as definições acima, pode-se dizer que agentes inteligentes artificiais constituem um novo paradigma de concepção de sistemas computacionais, no qual cada sistema deve ser visto como capaz de perceber um ambiente através de sensores e agir de forma racional e autônoma sobre esse ambiente. Estes sistemas podem conter mais de um agente, comunicando-se entre si, a fim de alcançarem os objetivos para os quais foram projetados.

Os atributos essenciais de um agente são: *autonomia*, *habilidade social*, *reatividade* e *pró-atividade*. Outros atributos não-essenciais identificados foram: *mobilidade*, *veracidade* e *benevolência*. Para completar sua definição, os autores explanaram cada um dos termos envolvidos, baseando-se em princípios formulados por outros autores, como a seguir:

Autonomia: agentes operam sem a intervenção externa de humanos ou outras entidades e conseguem manter um certo grau de controle sobre suas próprias ações, bem como seus estados internos (CastelFranchi, 1995).

Habilidade Social: agentes realizam interação com outros agentes (humanos ou outras entidades) usando para isto alguma linguagem de comunicação (Genesereth *et al.*, 1994) (Finin, 1997), (Fipa, 1997).

Reatividade: agentes percebem seu ambiente e respondem de maneira adequada às mudanças que nele ocorrem. Esse ambiente pode ser um mundo físico, uma interface gráfica, uma coleção de outros agentes, a Internet, ou uma combinação desses elementos.

Pró-Atividade: agentes não apenas reagem ao ambiente, mas também são capazes de apresentar um comportamento dirigido por objetivos ao tomar iniciativas. Ser dirigido por objetivos significa *ser intencional* (Cohen *et al.*, 1990).

Mobilidade: é a capacidade ou habilidade de um agente mover-se através de uma rede. Conceito formalizado por James White em 1994.

Veracidade: assume-se que um agente não transmitirá uma informação falsa conscientemente. A questão, *conscientemente*, nos leva à situação em que um agente pode ser enganado e isso possui probabilidades de ocorrer.

Benevolência: assume-se que um agente não apresenta objetivo conflitante e que *sempre tentará realizar as tarefas que lhe forem solicitadas* (Rosenchein *et al.*, 1985), o que nos envia a parâmetros de racionalidade.

Outras características podem ser consideradas, como o uso de estados mentais, tais como *conhecimento*, *crenças*, *desejos*, *intenções* e *engajamento* (Rao *et al.*, 1991). ou a adição de estados emocionais, assunto este que é matéria de muita pesquisa atualmente.

Embora alguns autores mencionem as diferenças e semelhanças entre objetos e agentes, para o projeto Mathnet, a diferença relevante entre objetos e agentes é o modo como se comportam conceitualmente. Objetos podem ser acionados com uma “ordem” para a realização de uma tarefa, enquanto que para os agentes, realiza-se uma “solicitação” para que uma tarefa seja efetivada. Os agentes podem se recusar a realizar uma tarefa, enquanto que um objeto não possui esta capacidade de escolha. Este diferencial é importante e fundamental para separar a forma de pensar “*orientado a agente*” da forma de pensar “*orientado a objeto*” (Odell, 1999).

2.2.2 Agentes MATHNET

O Ambiente MATHNET é baseado em uma arquitetura Multiagentes composta por agentes inteligentes artificiais e por agentes humanos (Labidi et al., 2000b).

Na Figura 2.2 podemos visualizar os principais agentes que compõem o ambiente MATHNET. Além destes, outros agentes secundários também fazem parte do sistema. Um exemplo disto pode ser observado em (Nunes, 2001), que descreve o serviço de busca MATHNET baseado em agentes móveis.

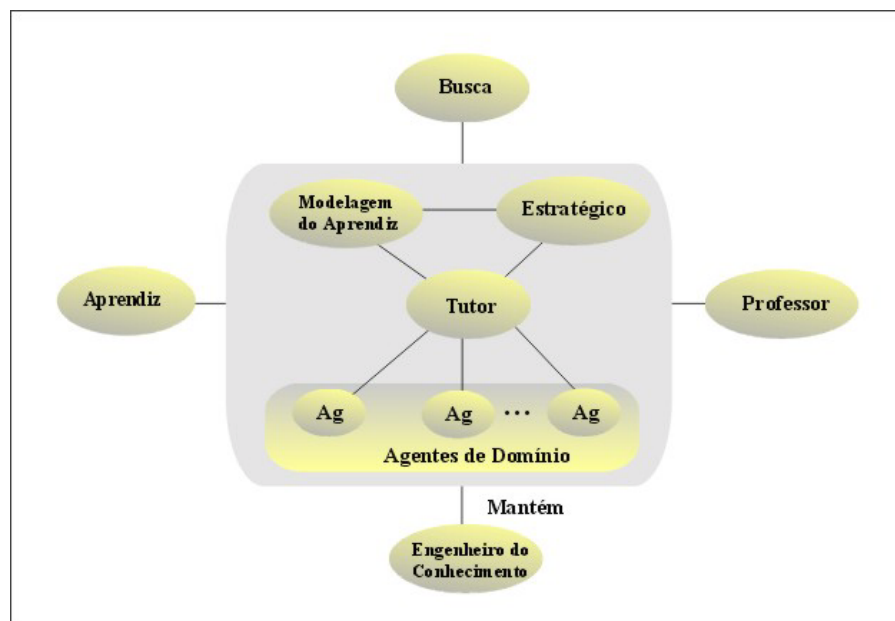


Figura 2.2 - Arquitetura Multiagentes MATHNET

A seguir veremos as características dos principais agentes do sistema MATHNET.

a) Agentes Artificiais

- **Agente Tutor:** Participa de todas as atividades de ensino e avaliação. Ele controla as interações dos grupos com o sistema durante o processo de ensino / aprendizagem. Também é responsável pela apresentação do conhecimento aos aprendizes. Realiza suas atividades de acordo com as estratégias definidas pelo professor e pelo Agente Estratégico e ainda interage com os Agentes de Domínio e o Agente Estratégico a fim de apresentar o conteúdo adequado para cada um dos aprendizes.

- **Agentes de Domínio:** Cada agente é detentor de um certo conhecimento, representando um especialista. Um Agente de Domínio é responsável pelo armazenamento e representação de um conhecimento específico (Labidi et al., 2000b), sendo esta a única característica que diferencia um Agente de Domínio de outro. Ele também pode ter links para recursos armazenados no servidor de conteúdos.

- **Agentes de Modelagem do Aprendiz:** É responsável pelo processo de aquisição, representação e manutenção de informações sobre alunos e grupos durante o processo de ensino / aprendizagem (Coutinho et al., 2000).

- **Agente Estratégico:** É responsável pela definição das estratégias pedagógicas a serem adotadas pelo Agente Tutor a partir da interação com o Agente Modelagem do Aprendiz e, após observar o comportamento de cada aprendiz, é quem decide o quê, quando e o conteúdo que será apresentado aos alunos.

- **Agente de Busca:** No Mathnet, os aprendizes possuem três alternativas para tirar dúvidas relativas ao que está sendo estudado: 1) consultando o banco de dúvidas; 2) interagindo entre si e com o professor através do uso de *chat*, *e-mail*, teleconferência, etc; 3) recuperando informações com o auxílio de um serviço de busca. Em (Nunes, 2001) é descrito o funcionamento do Serviço de Busca Mathnet, baseado no perfil do aprendiz e utilizando agentes móveis (Lange, 1998). O Serviço de Busca Mathnet tem como objetivo apresentar para o aprendiz informações personalizadas que possam ajudá-lo a tirar suas dúvidas. Este serviço é baseado em agentes de busca (Kotz, 1999), o qual instancia um conjunto de agentes inteligentes móveis que tentam acessar recursos de dados remotos. Este processo de recuperação de informações retorna

documentos ou *links* para documentos, levando em consideração o modelo do aprendiz.

b) Agentes Humanos

- **Agente Professor :** Agente humano que pode assumir diferentes papéis no sistema, destacando-se porém o de orientador, avaliador e especialista. A sua percepção quanto ao desenvolvimento dos estudantes é fundamental para lograr-se bom êxito na tarefa fundamental do Sistema MATHNET, que é ensinar. Entre outras funcionalidades, o professor poderá:

- i) Fornecer ao Engenheiro do Conhecimento o conteúdo a ser ministrado aos alunos;
- ii) Com o auxílio do sistema, formar grupos e reorganizá-los, quando for necessário;
- iii) Modificar a estratégia pedagógica adotada pelo Agente Tutor;
- iv) Supervisionar e interagir com as áreas cooperativas, monitorando a apresentação dos conteúdos, discutindo e esclarecendo dúvidas dos aprendizes;
- v) Avaliar os aprendizes, tendo como base as informações do modelo do aprendiz.

- **Agente Aprendiz:** São os alunos, os quais necessitam estar inseridos em grupos para participar das sessões de aprendizagem, podendo estar separados fisicamente ou não dos demais componentes do grupo. São os usuários alvo do sistema e o bom desempenho deles é a meta a ser alcançada.

- **Agente Engenheiro do Conhecimento:** É responsável por fazer a manutenção dos Agentes de Domínio, o que inclui a edição do conhecimento de cada um deles e a organização de seus domínios.

2.2.3 Núcleo Padrão de Agentes Artificiais MATHNET

O núcleo padrão dos agentes artificiais MATHNET é um conjunto de classes, embutidas na ferramenta ZEUS (Nwana et al., 1999), que provê toda funcionalidade de comunicação interagentes, planejamento e execução de tarefas.

A ferramenta ZEUS se caracteriza por ser um ambiente gráfico que fornece uma biblioteca de componentes de software e ferramentas que facilitam uma rápida construção e desenvolvimento de sistemas multiagentes. Ele é gratuito, de código aberto e constitui-se por um conjunto de componentes, escritos em Java, que podem ser descritos em três grupos funcionais (ou bibliotecas): biblioteca de componentes para agentes, uma ferramenta construtora de agentes e um conjunto de utilitários, incluindo um servidor de nomes, um facilitador e um visualizador de agentes.

A Figura 2.3 mostra a interface principal da ferramenta Zeus.

A biblioteca de componentes fornece, entre outras coisas, uma linguagem de comunicação entre agentes, baseada em performativas, neste caso FIPA-ACL. Além disso, ZEUS opera em redes TCP/IP, o que facilita a intercomunicação entre agentes executados em sistemas operacionais diferentes.

Além dos agentes artificiais principais da arquitetura MATHNET, existem outros agentes que fazem parte do núcleo, como os Agentes de Interface, por exemplo. Esses agentes são responsáveis por conduzir toda a conversação entre os agentes humanos e os agentes artificiais do sistema.

Os agentes do Serviço de Busca MATHNET (Nunes, 2001), por serem móveis, não são desenvolvidos no ZEUS, pois esta ferramenta não suporta mobilidade. Esses agentes são implementados usando a plataforma Aglets (A ferramenta IBM Aglets Software Development Kit está disponível em <http://www.trl.ibm.co.jp/aglets>) , a qual foi desenvolvida pelo laboratório de pesquisas da IBM.

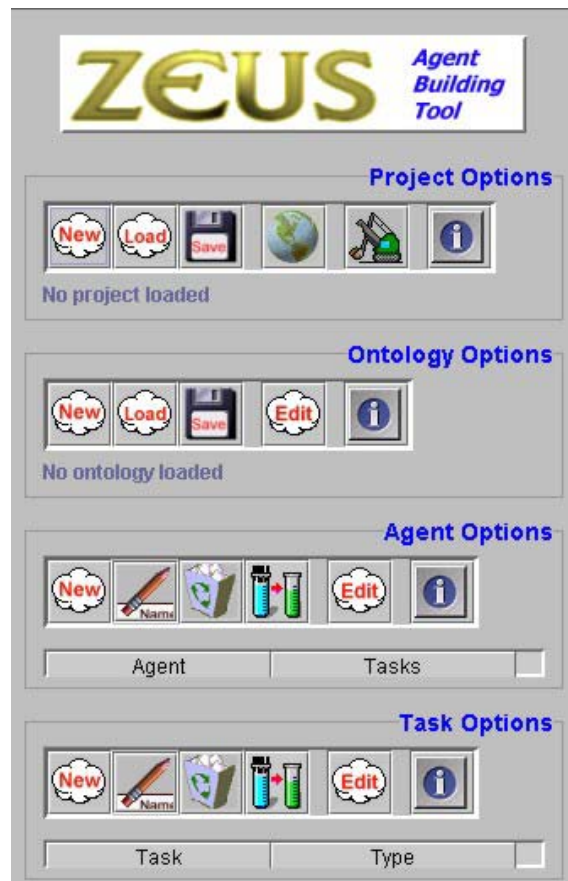


Figura 2.3 - Interface principal do ZEUS Agent Toolkit

2.3 Interatividade

Todos os conceitos da aprendizagem cooperativa ressaltam a importância das interações entre alunos, grupos e professores em busca da eficiência no processo de ensino-aprendizagem. No MATHNET, os agentes humanos utilizam os recursos do sistema para interagir entre si. Por sua vez, os agentes artificiais interagem entre si e com os agentes humanos, com o objetivo de dar suporte computacional à aprendizagem.

Dos vários tipos de interações entre os agentes, destacam-se as seguintes interações (Labidi et al., 2000a):

a) Agente Tutor – Alunos / Grupos: Os alunos interagem com o Agente Tutor durante todo o decorrer do curso. Por exemplo, o Agente Tutor poderá ter a iniciativa de fornecer

dicas aos aprendizes, sugerir discussões, etc. Os aprendizes podem pedir ao Tutor que repita parte específica de uma apresentação, pedir/aceitar ajuda, expor dúvidas, aceitar/rejeitar sugestões de interações inter e intragrupos, etc.

b) Agentes Artificiais – Professor: O professor pode interagir com vários agentes artificiais do MATHNET. Por exemplo, interage com o Agente de Modelagem do Aprendiz para solicitar informações sobre o aprendiz; com o Agente Tutor, para formar grupos; com o Agente Estrategista, para definir estratégias pedagógicas, etc.

c) Interação Intragrupo : Os alunos em um grupo cooperam para o alcance de objetivos comuns. As interações entre os componentes de um grupo podem ocorrer a qualquer momento, dentro e fora das sessões de aprendizagem, podendo a comunicação ser síncrona ou assíncrona. Essas interações podem ser importantes durante as atividades de assimilação e aplicação do conhecimento.

d) Interação Intergrupos: Visando eficiência na aprendizagem, a interação entre os grupos assume papel importante. Através deste tipo de interação, os alunos têm oportunidade de conhecer uns aos outros, trocar experiências e conhecimentos, etc.

e) Professor – Grupos / Alunos: Este tipo de interação é muito importante para o melhor andamento da aprendizagem, principalmente no caso de dúvidas não resolvidas pelos outros alunos, grupos ou agentes artificiais. A qualquer momento, o professor poderá enviar e receber mensagens dos alunos. Além disso, o MATHNET pode funcionar sem a presença *online* do professor. Neste caso, as mensagens destinadas/originadas pelo professor podem ser enviadas/recebidas fora da sessão de aprendizagem. Um exemplo disso pode ser observado na utilização do Banco de Dúvidas, o qual é um repositório de dúvidas compartilhado, onde os alunos cadastram suas dúvidas, colocando-as à disposição de todos os usuários do sistema, podendo ser respondidas tanto pelo professor quanto pelos outros alunos.

Muitas são as maneiras que um sistema de aprendizagem cooperativa tem para incentivar e propor interações entre aprendizes. O sistema PairSM (Bull et al., 1997) faz uma comparação com os modelos dos aprendizes e, baseado nos conceitos da Zona Proximal de Desenvolvimento (Vygotsky, 1978), infere quais conhecimentos que pares de aprendizes podem revelar quando trabalham juntos. Através desta inferência, o sistema poderá

recomendar interações entre pares de aprendizes de acordo com os conhecimentos que eles podem vir a desenvolver através de interações.

Em (Nakamura et al., 1996) são apresentados alguns métodos utilizados para estimular alunos à discussão em grupos, através da intervenção de um pseudo-aprendiz (artificial). Esses métodos utilizam o modelo do aprendiz para adaptar as intervenções:

i) *Responder uma declaração desprezada:* Quando a discussão torna-se muito ativa, muitas declarações feitas por alguns alunos “menos competentes” tornam-se desprezadas e tendem a não ser comentadas pelos outros. Um pseudo-aprendiz faz comentários sobre suas declarações, encorajando-os.

ii) *Fingir não conhecer a resposta:* Em um grupo de aprendizagem, os alunos tendem a depender de um aprendiz mais hábil que todos os outros. O sistema simula um pseudo-aprendiz hábil. Portanto, quando um aluno “menos competente” faz perguntas a esse pseudo-aprendiz, ele não responde a algumas perguntas e nem as encaminha para os aprendizes capazes de respondê-las. Isso estimula as discussões em busca das respostas.

iii) *Falar ativamente:* Quando um aprendiz não está discutindo com os outros, um pseudo-aprendiz “fala” ativamente com ele, através da interface do aprendiz, abordando vários assuntos relacionados ao que está sendo estudado. Para isso, deve-se levar em conta os conhecimentos já assimilados pelo aprendiz. Isso o estimula a participar das discussões.

iv) *Sustentar uma opinião minoritária:* O sistema, através de um pseudo-aprendiz, comenta opiniões de alunos menos *competentes* que a maioria discorda ou não quer comentar. Isso incentiva aprendizes mais tímidos a participar das discussões.

2.4 Atividades Pedagógicas

As atividades de ensino-aprendizagem, no sistema MATHNET, são classificadas em seis tipos (Figura 2.4). Cada tipo de atividade tem funções específicas que são desempenhadas com o uso de estratégias pedagógicas apropriadas, escolhidas de acordo com as especificações do modelo do aprendiz (Coutinho et al., 2000).

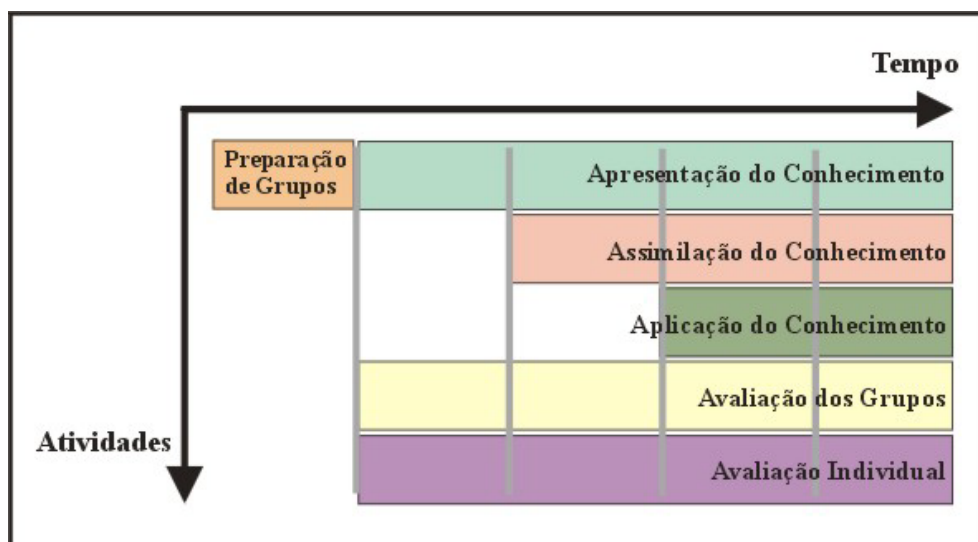


Figura 2.4 - Atividades Pedagógicas

Antes de iniciar as sessões de ensino-aprendizagem, o aprendiz preenche alguns formulários, fornecendo informações pessoais ao sistema, preferências e atributos relevantes para a formação dos grupos. Em seguida, é cadastrado no sistema e, finalmente, recebe instruções sobre utilização da interface a ele disponível. Após essas preliminares, o sistema dará início às atividades pedagógicas.

1. Preparação de Grupos: Ocorre sempre no início de uma sessão de aprendizagem, com a finalidade de preparar os grupos para as atividades de ensino-aprendizagem. Nesse momento, o professor, com o auxílio do sistema e baseado nas informações dos aprendizes, poderá organizar os alunos em grupos. Este tipo de atividade pode ser entendido como uma das fases de planejamento do curso. Essa preparação é indispensável para todas as sessões. As atividades subsequentes são típicas de ensino-aprendizagem e avaliação e a sua sequência depende da estratégia global estabelecida pelo professor.

Depois de realizadas as atividades concernentes à Preparação de Grupos, a sequência das atividades subsequentes depende da estratégia adotada pelo professor e o sistema, a fim de obter-se maior eficiência na aprendizagem.

2. Apresentação do Conhecimento: Corresponde à apresentação do conteúdo nos terminais dos alunos. O Agente Estratégico define o conteúdo e a apresentação mais adequada, com base no do perfil de cada estudante.

3. Assimilação do Conhecimento: Este é o primeiro tipo de atividade realmente cooperativa. Nesta atividade, os alunos interagem entre si com o objetivo de assimilar o conhecimento apresentado. O Agente Tutor poderá intervir nas interações se assim julgar necessário.

4. Aplicação do Conhecimento: Esta atividade é muito importante na fixação do conhecimento supostamente adquirido. Os alunos trabalham cooperativamente na realização de tarefas, como por exemplo, resolução de problemas (Coutinho, 1999).

5. Avaliação do Grupo: A atividade de avaliação é iniciada desde o momento em que os alunos estão assistindo à apresentação do conhecimento. Todas as atividades auxiliam o Ambiente MATHNET na composição da avaliação final. Na avaliação final de sessão (avaliação propriamente dita), os mesmos métodos utilizados na Aplicação do Conhecimento são também aqui utilizados com a finalidade de complementar o que o ambiente já conhece sobre o grupo.

A Avaliação Final de Sessão é realizada pelo Agente Tutor, com o objetivo de complementar a avaliação feita ao longo da sessão. Para isso, o Agente Tutor leva em consideração as informações presentes no modelo do grupo e alguns critérios pré-estabelecidos pelo professor no início da sessão. Os métodos utilizados na Aplicação do Conhecimento podem ser aqui utilizados.

6. Avaliação Individual: O sistema realiza a avaliação individual de forma semelhante à Avaliação do Grupo, com a diferença de que o sistema avalia o aluno individualmente e não em grupo. Portanto, as tarefas são apresentadas ao aluno, individualmente, e o modelo do aprendiz individual é consultado para dar suporte à avaliação e não existe interação dos alunos com outros ou com os agentes.

2.5 Estratégias Pedagógicas

Mesmo que as pessoas possuam o mesmo nível de conhecimento, normalmente, o estilo de aprendizagem de cada um é diferente. Alguns aprendem mais facilmente participando de atividades de discussão; outros, por meio de observações; outros através de exemplos, analogias, etc. Para adaptar o ensino ao estilo de aprendizagem de cada aluno ou grupos, são utilizadas as estratégias pedagógicas. Estratégias pedagógicas são métodos e técnicas pedagógicos utilizados pelo professor, objetivando uma maior eficiência no processo de ensino-aprendizagem, frente às diferenças comportamentais dos aprendizes.

Tradicionalmente, um professor pode mudar a estratégia pedagógica utilizada, caso perceba que o assunto está sendo ensinado de forma não muito eficiente. Ele pode fornecer explicações, introduzir exemplos, desafiar o aprendiz com questões, omitir a parte introdutória do assunto, e muitas outras estratégias de ensino que são utilizadas no dia-a-dia de um professor. Essa flexibilidade do professor para mudar a estratégia pedagógica tem fornecido eficiência na condução do ensino (Spensley, 1990). Mas, para esse professor, a adaptação das estratégias pedagógicas ao estilo de cada aluno, somente é possível se a quantidade de aprendizes for relativamente pequena, pois ele necessitaria obter muitas informações para adequar-se às particularidades pessoais e estado de conhecimento de cada aluno. A quantidade de alunos pode ser aumentada com a informatização desse processo, através do uso de Sistemas Tutores Inteligentes (STI).

Apesar de os STI possuírem capacidade de interagir com muitos aprendizes ao mesmo tempo, tradicionalmente, um STI apresenta pouca flexibilidade a respeito do uso de estratégias pedagógicas e, normalmente, é desenvolvido seguindo uma estratégia fixa que seria aplicada a todos os alunos (Abou-Jaoude et al., 1998). Para que um STI interaja de forma eficiente e personalizada com os aprendizes, é necessário “imitar” ao máximo a capacidade que o professor possui na mudança de estratégias pedagógicas.

O MATHNET possui a possibilidade de múltiplas estratégias pedagógicas, como também introduz o conceito de Agente Estrategista que interage com o Agente de Modelagem do Aprendiz para selecionar a estratégia mais adequada de ensino com base no modelo do aprendiz.

Existem dois tipos de estratégias utilizados pelo Agente Estrategista MATHNET:

1) *Estratégia Global:* Estratégia definida ou modificada pelo professor nos intervalos entre as sessões de ensino-aprendizagem.

2) *Estratégias Específicas de Atividades:* Estratégias que podem ser modificadas pelo Agente Estrategista ou professor durante as atividades em uma sessão.

A seguir, são relacionadas algumas estratégias pedagógicas utilizadas no MATHNET, levando-se em consideração os dois tipos de estratégias que podem ser utilizados. No entanto, outras estratégias podem ser acrescentadas às já existentes. Inclusive, está sendo pesquisada a viabilidade de se incluir ao MATHNET um Editor de Estratégias Pedagógicas.

2.5.1 Estratégia Global

A Estratégia Global faz parte do planejamento pedagógico realizado antes do início de uma sessão de ensino-aprendizagem. Essa estratégia diz respeito à sequência das atividades pedagógicas e ao tempo alocado para a realização de cada uma delas. Portanto, nessa estratégia é definida:

i) *A sequência das atividades durante a sessão:* A sequência das atividades é muito importante para o bom andamento do processo de ensino-aprendizagem. Por exemplo, uma sessão poderá iniciar com uma atividade de assimilação de conhecimento, depois finalizar com as avaliações de grupo e individual;

ii) *O tempo de duração alocado para cada uma das atividades:* Para cada tipo de atividade é alocado um tempo para que a sessão tenha um limite de tempo máximo para terminar. Isso facilita o sincronismo entre as tarefas realizadas pelos grupos.

2.5.2 Estratégias Específicas de Atividades

Durante as atividades de ensino-aprendizagem e avaliação, muitas estratégias podem ser adotadas (Ferreira, 1998) (Frasson et al., 1997) (Frasson, 1998). No entanto, o uso destas limita-se aos recursos disponíveis em um ambiente de aprendizagem. Nesta pesquisa, foram identificadas algumas estratégias específicas que podem ser adotadas pelo Agente Estrategista MATHNET, conforme o tipo de atividade:

a) Estratégias de Apresentação

- Expositiva: Trata-se de uma aula convencional, porém apresentada com recursos multimídia no terminal disponível para cada um dos aprendizes;
- Argüição: Inicia a apresentação com questionamentos, seguidos de respostas acerca do conteúdo a ser ensinado. Suscita a curiosidade dos aprendizes e os motiva para uma participação mais intensa no decorrer da aprendizagem;
- Analogia: Consiste em estabelecer comparações entre o que será ensinado e o mundo real, facilitando assim a fixação do assunto por parte dos aprendizes;
- Contagem de histórias: Trata do assunto de forma simples e ajuda na rápida fixação por meio de situações relacionadas.

b) Estratégias de Assimilação

Na fase de assimilação, os aprendizes estarão interagindo, discutindo e trocando informações entre si, em busca da assimilação do conteúdo apresentado. O Agente Tutor poderá intervir na discussão, sugerindo uma questão para um aprendiz (aluno ou grupo), mostrando uma opinião oposta, levantando discussões com alunos pouco interativos, etc. (Nakamura et al., 1996).

c) Estratégias de Aplicação

- Negociação: Os estudantes trabalham juntos. Eles “negociam” uns com os outros durante todas as etapas do problema, a fim de solucioná-lo em conjunto, da melhor forma possível;

- Competição: Nesta estratégia, a mesma tarefa é alocada para todos os membros do grupo. Cada aprendiz tenta resolvê-la separadamente no menor espaço de tempo possível. Depois, interagem entre si para apresentar as respostas e expor suas dúvidas;
- Co-Ação: Semelhante à estratégia anterior, mas não há competição entre eles.
- Complemento: Permite que a tarefa alocada para o aprendiz represente uma pequena parte de uma tarefa maior. Assim, a realização do trabalho depende da performance na realização das sub-tarefas por parte de cada um.
- Assistência: Algumas tarefas são alocadas para determinados aprendizes. Após resolverem, explicam para os outros membros do grupo, os quais podem auxiliá-los quando necessário.

d) Estratégias de Avaliação de Grupo

Nesse tipo de atividades, as estratégias que podem ser utilizadas são as mesmas da Aplicação de Conhecimento. Nessa instância, o sistema propõe aos alunos a realização de tarefas a fim de completar a avaliação dos grupos.

e) Estratégias de Avaliação Individual

Na realização da avaliação individual, não existem muitas possibilidades de implementação de estratégias pedagógicas. Cada aprendiz se depara com tarefas individuais (ex: resolver uma lista de problemas de Física) e não pode interagir com os outros componentes de seu grupo e nem receber dicas do sistema para a realização destas.

2.5.3 Critérios de seleção das estratégias pedagógicas

Cada estratégia pedagógica possui vantagens peculiares no processo de ensino-aprendizagem. Elas devem ser escolhidas adequadamente, de acordo com o estilo de aprendizagem de cada aprendiz, visando à eficiência durante o processo de ensino. Para isso,

critérios devem ser estabelecidos com base no tipo de domínio, na forma de ensinar o domínio e nas informações que o sistema tem sobre os aprendizes.

a) Seleção da Estratégia Global - Quanto à Estratégia Global, o professor define a sequência das atividades e o tempo para cada uma delas de acordo com a forma de ensinar o assunto. Por exemplo, sendo Física, o assunto a ser ensinado em uma sessão de 1:40 minutos de duração, o professor poderá enfocar a aplicação do conhecimento da seguinte forma:

i) Apresentação do Conhecimento – 20 minutos: Após a preparação dos grupos, é iniciada a sessão com a apresentação do assunto em estudo;

ii) Aplicação do Conhecimento – 55 minutos: Aplica o conhecimento, propondo para os alunos/grupos atividades relacionadas ao assunto apresentado;

iii) Avaliação de Grupos e Individual – 25 minutos: Avalia os grupos e alunos.

b) Seleção de Estratégias Específicas de Atividades - A escolha das Estratégias Específicas é responsabilidade do Agente Estrategista. No entanto, o professor poderá, a qualquer momento, modificar uma estratégia específica, caso discorde da decisão do Agente Estrategista em razão de perceber que o andamento do curso não está sendo satisfatório. O Agente Estrategista seleciona essas estratégias com base em dois critérios:

i) Domínio a ser ensinado: A escolha das estratégias está estritamente relacionada ao tipo de conteúdo a ser ensinado. Ao criar o modelo do domínio, o professor define preferências para determinadas estratégias específicas;

ii) Informações sobre o aprendiz: O Agente Estrategista utiliza informações contidas no modelo do aprendiz.

2.7 O processo de avaliação

Avaliar consiste em fazer um julgamento sobre resultados, comparando o que foi obtido com o que se pretendia alcançar (Haydt, 1997). Dessa forma, a avaliação pode ser útil para orientar tanto os alunos quanto o professor, de forma que:

- 1) Fornece informações ao aluno para melhorar sua atuação;
- 2) Dá elementos ao professor para aperfeiçoar seus procedimentos pedagógicos.

A avaliação não visa a punir os alunos, mas orientar o processo de aprendizagem. Nesse sentido, a avaliação permite ao aluno conhecer seus erros e acertos, auxiliando-o a corrigir suas falhas.

A aprendizagem, assim como qualquer processo psicológico, não pode ser medido ou avaliado de forma trivial. O que pode ser medido são alguns comportamentos que ajudam a inferir se houve ou não aprendizagem (Tyler, 1974). Por exemplo, resultados de resolução de problemas, frequência, quantidade e qualidade nas interações, etc.

No MATHNET, a avaliação não se restringe a apenas quantificar os resultados de alguns problemas propostos aos aprendizes. Ela consiste na coleta de dados quantitativos e qualitativos e na interpretação destes com base em critérios previamente estabelecidos pelo professor.

Nas atividades de Avaliação de Grupo e Avaliação Individual ocorre a Avaliação Final de Sessão. A Avaliação Final de Sessão é um complemento da avaliação realizada ao longo da sessão de aprendizagem, através da representação mantida pelo modelo do aprendiz. Portanto, a avaliação no MATHNET ocorre durante todo o processo de aprendizagem e o modelo do aprendiz é indispensável para a realização desta.

Em função de que na prática cada professor possui sua metodologia de avaliação, no MATHNET, é o professor quem define os parâmetros que serão levados em consideração no processo avaliativo. O professor define conceitos (Ótimo, Bom, Regular, Ruim) para cada aprendiz ou grupo de acordo com alguns critérios por ele estabelecidos. O professor pode definir determinados pesos para dados critérios, como por exemplo:

- i) Frequência durante as apresentações – peso 2;
- ii) Quantidade e qualidade nas interações – peso 3;
- iii) Conhecimento dos principais conceitos e habilidades envolvidos no assunto – peso 10;
- iv) Resolução de exercícios de avaliação – peso 5.

A partir dos critérios, o sistema poderá realizar o cálculo da avaliação e dispor, ao professor, o resultado em conceitos (Ótimo, Bom, Regular, Ruim). Por sua vez, o professor poderá tomar atitudes a partir desse resultado, como exemplo, apenas apresentá-lo ao aprendiz, ou ainda, enviar mensagens de congratulações para o aluno, promover ou não o aprendiz para cursos de nível mais elevado, reorganizar grupos, etc.

Ao longo das atividades pedagógicas, o sistema armazena várias informações sobre os aprendizes, que servirão de base para a avaliação dos alunos e grupos. Portanto, o Agente Tutor realizará a Avaliação Final de Sessão com base nos parâmetros de avaliação definidos pelo professor e nas informações sobre os aprendizes, as quais são representadas no modelo do aprendiz.

Neste processo, o Assistente de Resolução de Problemas, participa, com seus resultados como um dos critérios que geram a avaliação final dos alunos e grupos.

2.8 Considerações Finais

Com este capítulo, observamos a estrutura do projeto Mathnet, a definição de agentes de alguns autores e a definição utilizada pelo Mathnet. A arquitetura do sistema foi detalhada por meio das definições dos diversos tipos de agentes presentes no projeto. Abordamos também a metodologia pedagógica utilizada e ainda descrevemos o processo de avaliação adotado.

Já conhecendo o ambiente em que nossa proposta será inserida, no próximo capítulo especificaremos em qual contexto de problema trabalharemos, veremos definições de problemas e analisaremos alguns métodos utilizados para resolvê-los.

3. RESOLUÇÃO DE PROBLEMAS

A resolução de problemas é uma área de estudo relativamente pobre de material, pois ainda existem grandes dificuldades em distinguir os processos a serem utilizados, em desenvolver instrumentos que avaliem esses processos e elaborar métodos que auxiliem a capacidade de resolver problemas.

As dificuldades, nos casos dos sistemas educacionais, acontecem devido às muitas variáveis envolvidas neste processo, citando-se entre elas, o aluno, professor, tarefa, contexto, afetividade e também o nível de desenvolvimento do aluno. Deve-se destacar a emoção (ansiedade, medo de não conseguir, etc.) envolvida nesse processo que varia ao longo do tempo de resolução, os alunos iniciam a tarefa com entusiasmo e com o passar do tempo, as reações positivas diminuem e as negativas aparecem.

Os estudos realizados nesta área não contribuíram para o desenvolvimento de uma teoria genérica ou ao menos de um quadro organizador.

Os principais resultados das investigações (Rosa et al., 2001) foram que as heurísticas, gerais e específicas, podem ser ensinadas e aprendidas e ajudam a melhorar o desempenho dos alunos na resolução de problemas; que os conhecimentos anteriores parecem determinar o sucesso na hora da resolução, resolver problemas em um ambiente propício ajuda a melhorar o desempenho do aluno; algumas estratégias são mais utilizadas que outras e alguns alunos não conseguem utilizar as estratégias em outro contexto ou por um longo período de tempo. O aluno fixará melhor o conteúdo se a ele for dada uma aplicabilidade. Ensinar estratégias por ensinar, levará o aluno a uma mera memorização.

Um outro resultado interessante está no desempenho dos alunos pesquisados, que está longe do que seria desejável. Embora possuam um certo conhecimento, os alunos não sabem como utilizá-lo e não percebem qual é a pergunta do problema.

3.1 O que são problemas

Problema é qualquer situação em que se precisa encontrar uma solução, no mais variado contexto, seja um pneu furado do carro, uma torneira com vazamento ou uma máquina de costura sem linha, estes são exemplos que retratam o conceito de problema que é em si bastante amplo.

Os problemas que abordaremos estão restritos a problemas educacionais ou acadêmicos (Português, Matemática, Física, etc.), sendo ainda mais especificamente voltado para problemas de domínio formal, que são resolvidos através da aplicação de passos, e mesmo neste domínio estaremos unicamente abordando problemas da Física.

A palavra “problema” normalmente é empregada de forma equivocada nas salas de aula. Geralmente, chama-se de problemas, uma série de exercícios aplicados pelo professor, que necessitam da aplicação rotineira de um procedimento já estabelecido: os exercícios que geralmente consistem em trabalhar certo número de atividades idênticas ou quase idênticas às que o professor usou de exemplo em sala de aula. Esta rotina é válida para a fixação, mas não é suficiente para compreensão plena a respeito de resolução de problemas.

Por causa disto, devemos atentar para que as atividades de resolução de problemas apareçam também como situações novas para os indivíduos que irão resolvê-las e com isto gerar oportunidades aos estudantes para que eles realmente resolvam problemas.

Especificamente, um problema é toda situação que requer a descoberta de informações desconhecidas para a pessoa que tenta resolvê-lo, e/ou a invenção de uma demonstração de um resultado dado. O fundamental é que o resolvidor tenha de inventar estratégias e criar idéias; ou seja: pode até ocorrer que o resolvidor conheça o objetivo a chegar, mas só estará enfrentando um problema se ele ainda não tem os meios para atingir tal objetivo.

Várias características de problemas foram apontadas por Resnick em (Davis, 1995) que resumidamente relatamos a seguir:

- **Sem algoritmização:** o caminho da resolução é desconhecido, ao menos em boa parte.
- **Complexos:** precisam de vários pontos de vista.

- **Exigentes:** a solução só é atingida após intenso trabalho mental; embora o caminho possa ser curto, ele tende a ser difícil.
- **Exigem lucidez e paciência:** para, na aparente desordem, vermos as regularidades, os padrões que permitirão a construção do caminho até a solução.
- **Nebulosos:** pode ocorrer que nem todas as informações necessárias estejam aparentes; por outro lado, pode ocorrer que existam conflitos entre as condições estabelecidas pelo problema
- **Não há resposta única:** além de, normalmente, ocorrer de existirem várias maneiras de se resolver um dado problema, pode ocorrer de não existir uma melhor solução e até de não existir solução.

Breuker afirma em (Breuker et al., 1994), que problemas são normalmente concebidos como a diferença entre o estado atual e um estado ideal de uma situação, e um problema está bem definido quando um simples teste pode concluir se uma solução para ele é verdadeira.

A definição de problema é composta, basicamente, por dois estágios. O primeiro estágio tem como resultado a diferença ou conflito entre o estado atual e um estado ideal, sendo estes dois estados a entrada para identificar um problema. A entrada do segundo estágio corresponde aos conflitos identificados na etapa anterior e deve gerar uma solução genérica, que definirá o problema.

3.2 Tipos de problemas

Usando a abordagem definida no CommonKADS (Breuker et al., 1994), que é uma metodologia de aquisição de conhecimentos que cobre o processo completo de desenvolvimento de sistemas baseados em conhecimento, pode-se classificar alguns tipos de problemas como os que seguem na Figura 3.1, atentando-se que uma situação pode apresentar características de mais de um destes tipos especificados ou até mesmo não se enquadrar em nenhuma destas especificações.

Síntese	Modificação	Análise
1. Modelagem 2. Design 3. Planejamento	4. Atribuição	5. Previsão 6. Monitoração 7. Diagnose 8. Avaliação

Figura 3.1 – Tipos de Problemas (CommonKads)

1. Modelagem: Esta classe de problemas está relacionada com a definição de qual é a interface com o ambiente. A conclusão genérica de uma modelagem é uma descrição comportamental (modelo) do sistema.

2. Design: É conceituado como um problema de geração de um projeto solução para algumas necessidade e desejos. Tem como conclusão uma estrutura de componentes e suas conexões. Os componentes podem ser objetos, processos físicos ou simbólicos.

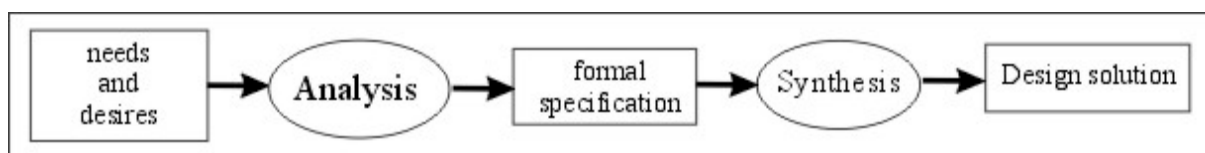


Figura 3.2 – Processo de Design (Breuker et al., 1994. p. 181)

3. Planejamento: São problemas onde um plano é construído, gerado, baseado em entradas de um estado inicial e de um estado desejado (meta). O estado inicial descreve a situação antes da execução do plano, enquanto o estado meta descreve obrigatoriamente aspectos da situação após a execução do plano. O plano é o resultado de um problema de planejamento.

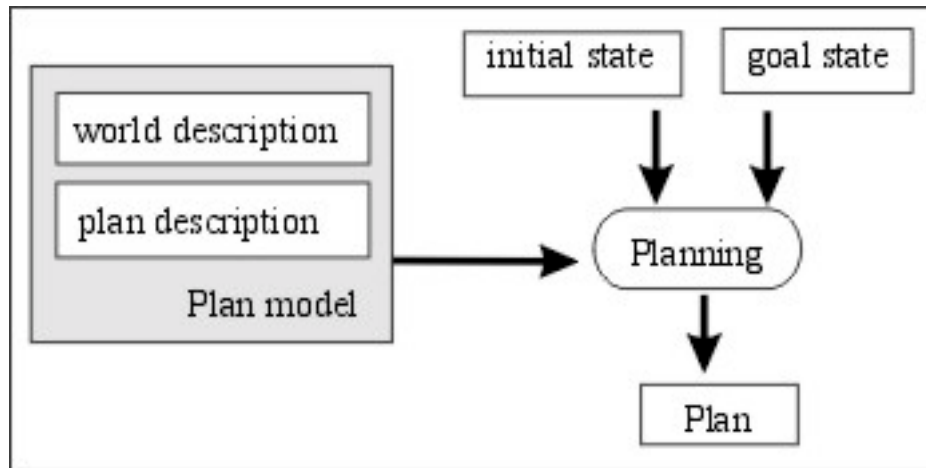


Figura 3.3 – Processo de Planejamento (Breuker et al., 1994. p. 214)

A descrição da situação contém os conhecimentos sobre a situação (domínio) no qual o plano irá tomar lugar, e a descrição do plano é a especificação de como as partes de um plano (ações, sub-planos) serão montadas juntas.

4. Atribuição: Estão relacionados com a distribuição de elementos (ações, componentes) sobre uma estrutura. Se a estrutura for um plano, é conhecido como agendamento, mas se a estrutura for um projeto, é conhecido como configuração. Funciona com, pelo menos, dois conjuntos (tipos) de objetos. Sua solução consiste no estabelecimento de relações de atribuição entre objetos de diferentes conjuntos, de tal modo que seus requisitos e restrições sejam satisfeitos.

Neste tipo de problema temos como entrada, conforme Figura 3.3, os componentes que são um conjunto de objetos que devem ser atribuídos para um recurso. Os recursos são um conjunto de objetos para o qual os componentes devem ser atribuídos. E como saída do processo de Atribuição, é gerado um conjunto de tuplas de atribuição, constituindo um sub-conjunto do produto cartesiano do conjunto de componentes e o conjunto de recursos, os quais satisfazem requisitos e restrições especificados.

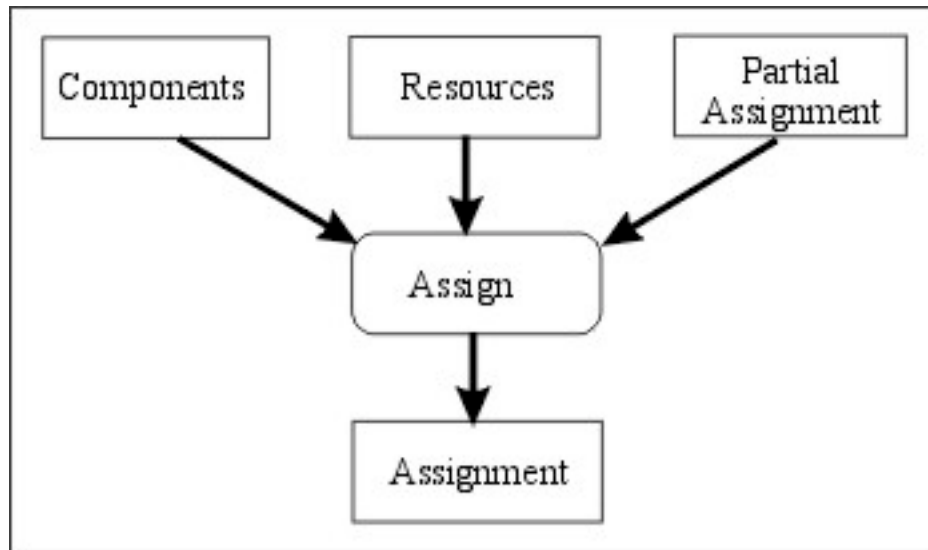


Figura 3.4 – Processo de Atribuição (Breuker et al., 1994. p. 232)

5. Predição: A partir de um estado inicial, estes problemas devem produzir antecipadamente à realidade um resultado final para um sistema. Ou seja, dado uma entrada, o resultado deste tipo de problema é uma previsão do que deve acontecer na realidade.

6. Monitoração: São problemas que produzem a diferença entre um estado predito e um estado observado.

7. Diagnose: Refere-se a problemas onde se encontram componentes ou estruturas que conflitam com seu modelo comportamental ou projeto.

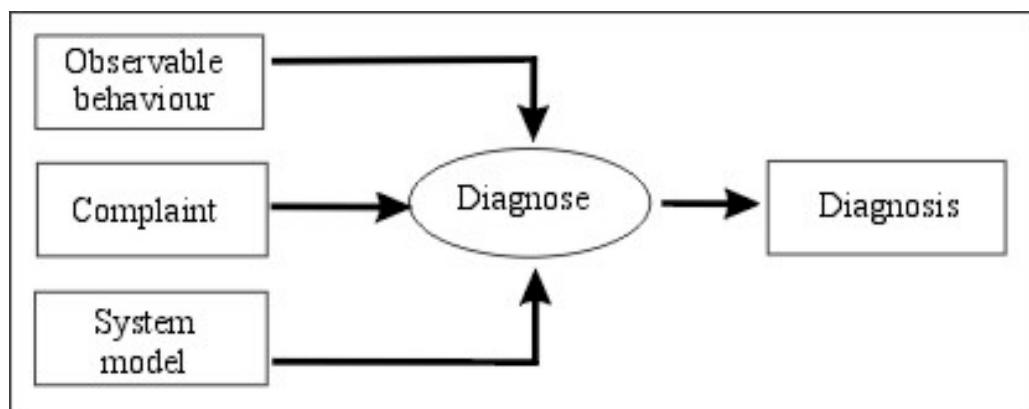


Figura 3.5 – Processo de Diagnose (Breuker et al., 1994. p. 138)

8. Avaliação: Diz respeito a problemas que provêm uma medida que classifica o comportamento de acordo com normas. São problemas onde uma descrição do caso (entrada) é mapeada para uma decisão (saída) de acordo com um modelo de sistema (entrada).

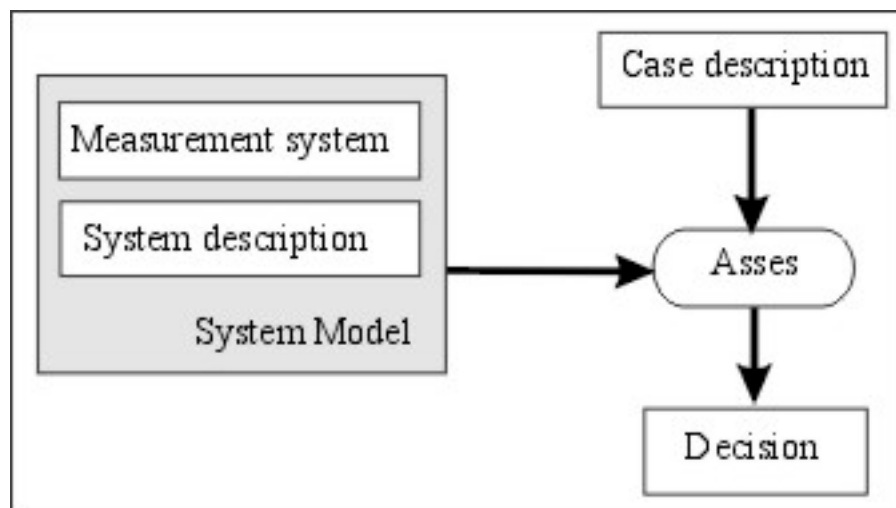


Figura 3.6– Processo de Avaliação (Breuker et al., 1994. p. 159)

O tipo de problema abordado pelo Assistente de Resolução de Problemas envolve características de três destes tipos: Modelagem, Planejamento e Atribuição, sendo esta definição baseada no propósito e escopo do Assistente. Comportando-se, segundo a definição do CommonKads de forma híbrida, o que certamente aumenta o grau de dificuldade de sua solução.

3.3 Métodos para resolução de problemas

Ao procurar-se uma solução para um problema, o ponto de vista, que é a maneira de encarar o evento, pode variar continuamente e é necessário variar esta posição em relação às diferentes situações apresentadas. A concepção do problema é, provavelmente, muito incompleta no início da abordagem, a perspectiva é transformada quando se alcança algum progresso e é mais diferente ainda quando se está chegando a uma solução.

3.3.1 Método de George Pólya

George Pólya teve uma carreira científica que se prolongou por mais de sete décadas. Ele deu boas contribuições em muitos domínios, principalmente na Matemática. Pólya (conforme Pólya, 1945), acreditava que existe uma arte da descoberta e também que a capacidade de descobrir e a capacidade de inventar podem ser desenvolvidas através de um ensino habilidoso que alerte os estudantes para os princípios da descoberta e que lhes dê oportunidade de praticarem estes princípios.

Pólya (em Pólya, 1945) sugere um processo de resolução de problemas dividido em quatro etapas, como pode ser visto na Figura 3.1, constituídas por algumas indagações cruciais na determinação do próximo passo da resolução, que serão mais bem explanadas posteriormente:

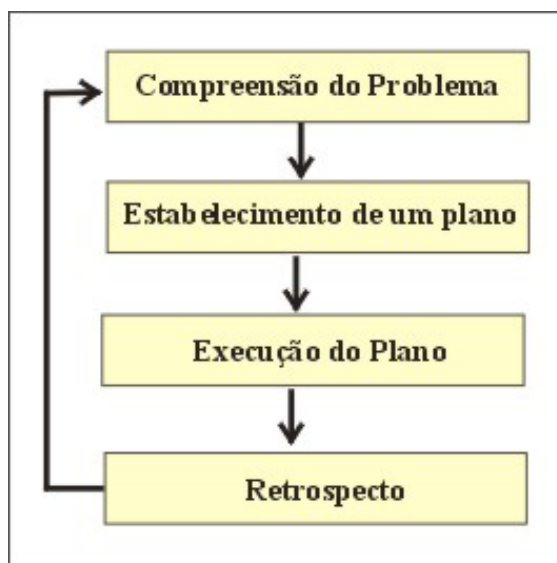


Figura 3.7 – Método do Pólya

Inicialmente, deve-se perceber claramente qual o objetivo do problema, que diz respeito ao entendimento do que está propondo o problema. Em um segundo instante, é preciso entender como os diversos itens estão inter-relacionados, como a incógnita está ligada aos dados, para então se ter uma idéia da resolução de estabelecer-se um plano. Na terceira etapa, executa-se o plano proposto e, finalmente, faz-se um retrospecto da resolução completa, revendo-a e a discutindo-a.

Cada fase tem uma importância específica no processo. Idéias brilhantes podem acontecer e, rapidamente, sem considerar todas as etapas, chegar-se à solução. O problema acontece quando uma idéia apenas parece brilhante, mas de fato e verdade não o é, e leva o estudante a uma solução inconvenientemente desastrosa e incorreta. Em vista desta grande possibilidade, o fato de não considerar qualquer uma das etapas, sem ter noção exata do que ela representa, é totalmente desestimulado.

Geralmente, acontecem erros quando um aluno, simplesmente, se coloca a fazer cálculos ou considerações, sem ter compreendido o problema. Normalmente, é pouco útil simplesmente executar detalhes, sem perceber a conexão principal ou sem ter feito um plano. Ainda com a verificação exaustiva de cada passo, muitos enganos podem ser evitados.

As fases são definidas a partir de alguns questionamentos a respeito do problema proposto, como veremos a seguir:

a) Primeira Fase - Compreensão do problema

Qual é a incógnita? Quais são os dados? Qual é a condicionante? É possível satisfazer a condicionante? A condicionante é suficiente para determinar a incógnita? Ou é insuficiente? Ou é redundante? Ou é contraditória?

Todas as indagações acima balizam a primeira fase na resolução de problemas. Quando se pode respondê-las adequadamente, chegou-se ao estado da compreensão do problema.

Uma pessoa pode ser denominada de tolo por responder a uma questão que não foi compreendida e também é frustrante trabalhar em algo para ter um fim indesejado. Estas “tolices” acontecem com uma certa frequência na resolução de problemas. Por isso, a identificação das respostas para os questionamentos desta fase conduz o estudante a uma alta probabilidade de chegar a uma solução correta.

O professor possui a tarefa de evitar que estas “tolices” aconteçam em relação ao conteúdo que ministrar. O aluno precisa compreender o problema, mas não somente isto, precisa também querer resolvê-lo. A falta de compreensão ou de interesse, nem sempre é

culpa do estudante. O problema deve ser bem escolhido, nem muito fácil, nem por demais difícil, e algum tempo deve ser destinado à sua apresentação, de forma natural e interessante.

O enunciado verbal precisa ficar bem entendido, pois, a partir dele, o aluno deve ter condições de identificar as partes principais do problema, a incógnita (é aquilo que se procura ou de que se precisa), os dados (são os valores iniciais explícitos no enunciado) e a condicionante (é o elo que une a incógnita de um problema aos respectivos dados). O estudante deve considerar estas partes atenta e repetidamente, sob vários pontos de vista, visando a alcançar o melhor entendimento possível.

Para ilustrar alguns pontos tratados acima, usaremos o seguinte exemplo: ***Calcular a diagonal de um paralelepípedo retângulo do qual são conhecidos o comprimento, a largura e a altura.***

Os estudantes, para discutirem com proveito este problema, precisam não apenas conhecer o teorema de Pitágoras e algumas das suas aplicações à Geometria Plana, mas também precisam ter um conhecimento superficial de Geometria Espacial, o que já seria suficiente.

O professor pode tornar o problema atraente, concretizando-o. A sala de aula seria um paralelepípedo retângulo cujas dimensões podem ser medidas ou estimadas. Os alunos devem calcular, “medir indiretamente”, a diagonal da sala. O professor poderia animar a figura que traçou no quadro-negro com diversas referências à sala de aula.

Um diálogo entre o professor e seus alunos poderia iniciar desta forma:

- 1- Qual é a incógnita? *O comprimento da diagonal de um paralelepípedo.*
- 2 - Quais são os dados? *O comprimento, a largura e a altura do paralelepípedo.*
- 3 - Use uma notação adequada. Qual a letra deve denotar a incógnita? *x*
- 4 - Quais as letras devem ser escolhidas para representar o comprimento, a largura e a altura?
 a, b, c
- 5 - Qual a condicionante que relaciona a, b e c com x? *x é a diagonal do paralelepípedo no qual a, b e c são, respectivamente, o comprimento, a largura e a altura.*

6 - A condicionante é suficiente para determinar a incógnita? *Sim, conhecendo a , b e c , conhece-se o paralelepípedo. Estando o paralelepípedo determinado, a sua diagonal também o ficará.*

b) Segunda Fase - Estabelecimento de um plano

Pode-se dizer que temos um plano quando conhecemos pelo menos de um modo geral, quais as contas, os cálculos ou os desenhos que são necessários executar para conhecer-se a incógnita. O caminho que vai desde a compreensão do problema até a formulação de um plano pode ser muito longo e tortuoso, visto que, a concepção da idéia de um plano é o principal feito na resolução do problema.

Esta idéia de um plano pode surgir gradativamente com a análise do problema ou repentinamente, após algumas tentativas frustradas, como uma idéia brilhante. O grande papel do professor ou a melhor forma de agir neste momento é propiciar, estimular o aprendiz, de forma discreta, a ter uma idéia luminosa.

As boas idéias são baseadas em experiências passadas e em conhecimentos previamente adquiridos, contudo, para se ter uma boa idéia, não basta apenas a simples recordação. Mas também não se pode ter uma boa idéia sem lembrar alguns fatos pertinentes. Exemplificando: apenas possuir os materiais para construir uma casa não é o suficiente para que ela seja feita, mas sem utilizar os materiais, é impossível fazê-la.

Os materiais indispensáveis para a resolução de um problema são certos itens relevantes do conhecimento já adquiridos, tais como problemas anteriormente resolvidos ou teoremas anteriormente demonstrados. Assim, é muito proveitoso começar uma resolução de problemas, indagando: *Conhece um problema correlato?*

A dificuldade desta abordagem se dá pelo fato de existirem, geralmente, muitos problemas relacionados com o nosso, tendo com este algum ponto em comum, e a dificuldade será em escolher aquele, ou os poucos que realmente são úteis.

Uma opção interessante é considerar um ponto comum essencial, como a incógnita e assim sendo a indagação reformulada seria: *Conhece um problema correlato com a mesma incógnita ou outra que seja semelhante?*

As indagações, quando bem compreendidas, muitas vezes contribuem para o início de uma seqüência de idéias corretas, mas não existe garantia de que as indagações dêem sempre resultados positivos. Neste caso, se faz necessário uma reavaliação para encontrar um outro ponto de contato. Temos que variar, transformar o problema. Daí, surge o questionamento: *É possível reformular o problema?*

Podemos para isto utilizar meios específicos da Variação do Problema, tais como a Generalização, a Particularização, o recurso à analogia, o abandono de uma parte da condicionante. Os detalhes são de extrema importância, mas não podem ser analisados agora. A variação do problema pode levar a um problema auxiliar adequado.

Na tentativa da aplicação de diversos problemas ou teoremas conhecidos, utilizando-se de diversas modificações e ensaiando problemas auxiliares diferentes, corre-se o risco de nos distanciarmos tanto do problema original que podemos chegar a perdê-lo por completo. No entanto, podemos utilizar uma boa indagação para voltarmos a ele: *Utilizou todos os dados? Utilizou toda condicionante?*

Voltando ao ponto em que paramos no exemplo anterior, os alunos haviam compreendido o problema e a expectativa do professor é que eles tomem uma decisão, tenham a iniciativa em definir um plano. Se o professor perceber que nenhuma atitude será tomada, deverá repetir cuidadosamente todo o seu diálogo com os estudantes, modificando as indagações, visando a uma melhor compreensão.

Considerando-se a possibilidade do silêncio por parte dos alunos, quando não souberem uma resposta (que será indicado por reticências), este novo diálogo poderia ser da seguinte forma:

1 – Conhece um problema correlato?

2 – Considerando a incógnita, conhece um problema que tenha a mesma incógnita ou outra semelhante?

- 3 – Qual é a incógnita? *A diagonal de um paralelepípedo.*
- 4 – Conhece um problema que tenha a mesma incógnita? *Não. Ainda não resolvemos nenhum problema que entre a diagonal de um paralelepípedo.*
- 5 – Preste atenção: a diagonal é um segmento de reta. Nunca resolveu um problema cuja incógnita fosse o comprimento de uma linha? *Sim, por exemplo, calcular o lado de um triângulo retângulo.*
- 6 – Muito bem. Eis um problema correlato já resolvido. É possível utilizá-lo?
- 7 – Temos um problema relacionado com o seu e já foi resolvido anteriormente. É possível utilizá-lo? É possível introduzir algum elemento auxiliar para possibilitar a sua utilização?
- 8 – O problema que foi resolvido se refere a um triângulo, analisando o seu problema, existe algum triângulo nele?

Se estas indagações forem insuficientes, devem continuar até o aluno ser despertado e chegar a um plano, mas com o último questionamento, espera-se que ele seja bastante explícito para dar a idéia da solução, que é a introdução de um triângulo retângulo, do qual a diagonal pedida é a hipotenusa, usando para esta conclusão o teorema de Pitágoras.

c) Terceira Fase - Execução do Plano

A concepção de um plano, a idéia da resolução, não é uma coisa simples de se conseguir. Para isto, é preciso conhecimentos anteriores, ter bons hábitos mentais, de concentração no objetivo e, certamente, um pouco de sorte. Executar o plano é muito mais fácil, mas é preciso que se tenha muita paciência.

O plano proporciona um roteiro geral e é preciso verificar se todos os detalhes estão inseridos neste roteiro, e esta verificação consiste em examiná-los, um após o outro,

pacientemente, até ficar tudo absolutamente claro e não exista nenhuma possibilidade de um erro estar oculto.

O grande risco nesta fase é o aluno esquecer o seu plano e isto acontece facilmente quando o estudante recebe o plano de fora, influenciado pelo professor ou alguém, sem de fato discernir o que foi desenvolvido. Entretanto, quando o aprendiz, mesmo que com ajuda externa, houver concebido o plano, não perderá com facilidade esta idéia.

Desta maneira, as indagações plausíveis para esta fase são:

É possível perceber claramente que o passo está correto?

Pode-se demonstrar que o passo está correto?

Retomando o problema exemplo do ponto em que deixamos, consideremos que o aluno, finalmente, teve a idéia da resolução.

O estudante percebe o triângulo do qual a incógnita x é a hipotenusa e a altura dada c é um dos catetos; o outro cateto é a diagonal de uma face. O professor deve insistir para que seja adotada uma notação apropriada. Pode-se escolher o y para denotar o outro cateto, que é a diagonal da face cujos lados são a e b . Desta forma, percebe-se com maior clareza a idéia da resolução, que consiste em introduzir um problema auxiliar cuja incógnita será y . Finalmente, calculando os dois triângulos, terá:

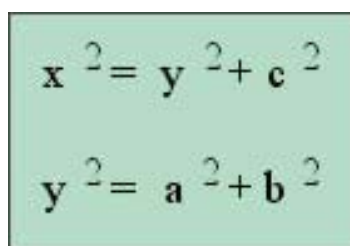

$$\begin{aligned}x^2 &= y^2 + c^2 \\ y^2 &= a^2 + b^2\end{aligned}$$

Figura 3.8 – Equação simples

A partir daí, eliminado-se a incógnita auxiliar y temos:

$$x^2 = a^2 + b^2 + c^2$$

$$x = \sqrt{a^2 + b^2 + c^2}$$

Figura 3.9 – Equação substituída

O professor não precisará interromper o aluno se este executar corretamente as operações, a não ser, possivelmente, para alertá-lo de que deverá verificar cada passo. Assim, ter-se-ia a seguinte argumentação: *É possível perceber claramente que o triângulo de lados x , y e c é retângulo?* Se a resposta positiva do aluno não for suficientemente convincente de continuar inquirindo: *Pode demonstrar que o triângulo retângulo?* Mas esta indagação pode ser suprimida até que a turma tenha uma boa base destes conhecimentos.

d) Quarta Fase – Retrospecto

Uma vez descoberta a solução do problema e escrita a demonstração, até mesmo os bons alunos dão por encerrado aquele assunto e partem para outro. Agindo desta maneira, uma instrutiva e importante fase do trabalho de resolução é perdida.

O simples fato de fazerem um retrospecto da resolução completa, reconsiderando e re-examinando o resultado final e os caminhos que os conduziram até esta resposta, ajudará grandemente na consolidação do seu conhecimento e aperfeiçoará a sua capacidade de resolver problemas.

O professor deve se preocupar em transmitir aos seus alunos o conceito de que problema algum fica completamente esgotado. Certamente, existe algo que ainda pode ser feito e, com estudo e aprofundamento, pode-se melhorar qualquer resolução, sabendo-se que, de qualquer maneira, é sempre possível aperfeiçoar a nossa compreensão da resolução.

Nesta fase, após um árduo trabalho, o aluno considera que resolveu corretamente o problema. Todavia, apesar do grande esforço, é sempre possível haver erros e, em especial, quando o argumento é longo e trabalhoso.

Devido a esta possibilidade, é conveniente o uso de verificações que podem ser definidas pelas seguintes indagações:

É possível verificar o resultado?

É possível verificar o argumento?

É possível chegar ao resultado por um caminho diferente?

Quando fazemos o retrospecto de uma solução, surge uma oportunidade natural de investigar as relações do problema. Desta feita, algo mais poderá ser conseguido como resultado do esforço de resolver este problema. O aluno poderá imaginar outras situações em que poderá utilizar os métodos adotados ou o resultado obtido.

Na última etapa do problema exemplo, os alunos haviam chegado à solução do problema: se as três arestas de um paralelepípedo retângulo, que se originam em um mesmo vértice, e são ***a***, ***b*** e ***c***, a diagonal será:

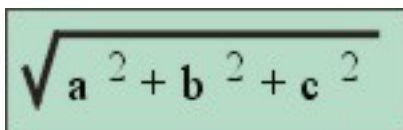

$$\sqrt{a^2 + b^2 + c^2}$$

Figura 3.10 – Equação 3

É possível verificar o resultado? O professor não pode esperar de um aluno inexperiente uma boa resposta para esta indagação. Mas é necessário que os alunos aprendam que os problemas “literais” apresentam uma grande vantagem sobre os puramente “numéricos”: se o problema for literal ele se prestará a diversas verificações, as quais não podem ser aplicadas a um problema numérico. O problema exemplo, embora simples, pode demonstrar esta propriedade. O professor pode apresentar diversas indagações que facilmente serão respondidas SIM, mas um NÃO revelará uma séria falha no resultado, tais como:

1 – Utilizou todos os dados? Todos os dados aparecem na fórmula que exprime a diagonal?

*2 – O nosso problema é de Geometria Espacial: calcular a diagonal de um paralelepípedo de dimensões dadas ***a***, ***b*** e ***c***. Ele é análogo a outro problema da Geometria Plana: calcular a*

diagonal de um retângulo de dimensões dadas, a e b . O resultado do nosso problema “espacial” será análogo ao resultado do problema “plano”?

3 – Se a altura c crescer, a diagonal também crescerá? A fórmula adotada mostra isto?

4 – Se todas as três dimensões do paralelepípedo crescerem em uma determinada proporção, a diagonal crescerá nesta mesma proporção. Se, na fórmula adotada, substituir-se a , b e c por $12a$, $12b$ e $12c$, respectivamente, a expressão da diagonal, devido a essa substituição, também deverá ser multiplicada por 12. Está certa esta afirmação?

Estas indagações e outras mais produzem diversos bons efeitos. O primeiro é o aluno perceber que a sua fórmula passou em tantos testes. Desta forma, estará convicto que sua fórmula está correta porque foi deduzida cuidadosamente e isto aumentará a probabilidade da fórmula não ser esquecida e com isto o conhecimento do aluno consolidar-se-á.

Finalmente, todas as indagações podem ser transferidas facilmente para problemas semelhantes. Após algumas experiências com problemas semelhantes, um estudante esforçado poderá perceber as idéias básicas gerais: a utilização dos dados relevantes, a variação dos dados, a simetria, a analogia. Adquirindo-se o hábito de dirigir a atenção para estes pontos, a capacidade de resolver problemas poderá definitivamente beneficiar-se.

A seguir, temos um exemplo simplificado de um problema resolvido pelo método do Pólya.

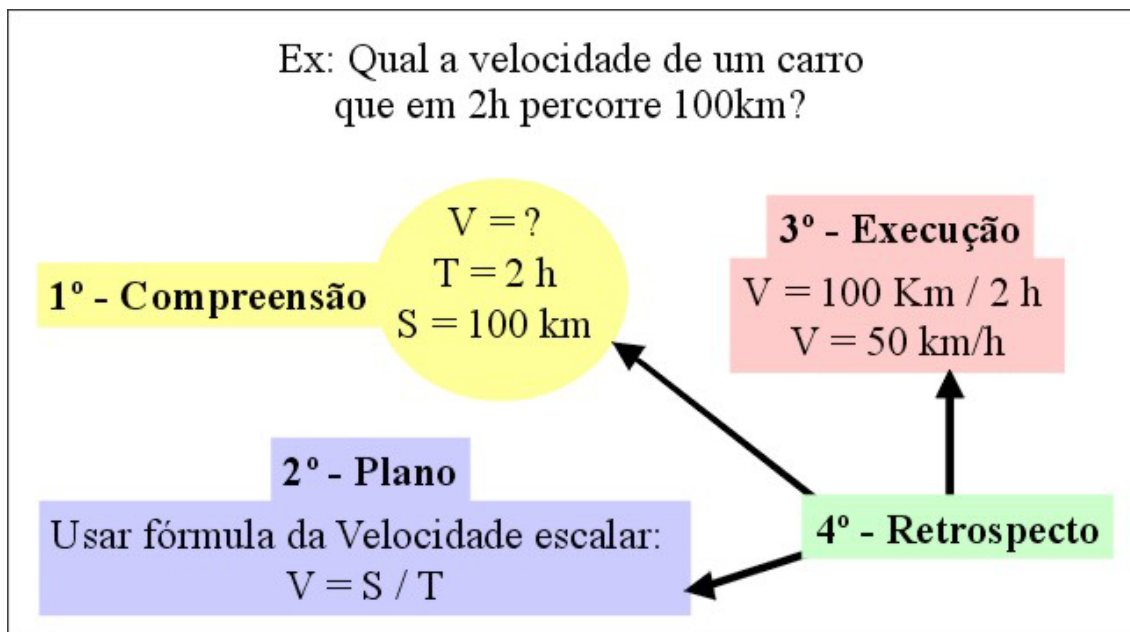


Figura 3.11 – Exemplo de Resolução de Problema pelo método do Pólya

3.3.2 Método do Serway

R. P. Feynman afirmou: “Você nada sabe antes de praticar”. Baseado nesta afirmação, Serway (Serway, 1996), que trabalha principalmente no domínio da Física, recomenda enfaticamente o exercício para poder-se resolver uma grande variedade de problemas. A capacidade de resolver problemas é um dos traços fundamentais do conhecimento da física. Para isto é essencial entender os conceitos e princípios básicos.

Um bom exercício consiste em procurar soluções alternativas para um mesmo problema. Por exemplo, os problemas de mecânica podem ser resolvidos pelas leis de Newton, mas muitas vezes também é possível uma solução mais direta mediante a considerações a respeito de energia.

O método de resolver problemas deve ser cuidadosamente organizado. Um plano sistemático é especialmente importante quando o problema envolve muitos conceitos.

Primeiramente, deve-se ler o problema cuidadosamente, várias vezes se for preciso, até ter certeza que entendeu o que foi perguntado. Procurar palavras-chaves que ajudem a

interpretar o problema ou levantar algumas hipóteses. A interpretação de um problema é parte integral de sua resolução.

É aconselhável adquirir o hábito de escrever as informações fornecidas em um problema e determinar as grandezas que precisam ser encontradas.

Serway observou que muitos estudantes não conhecem as limitações de certas fórmulas e leis da física, em situações particulares. Se faz necessário compreender e não esquecer as hipóteses subjacentes a uma determinada teoria ou a um certo formalismo. Por exemplo, algumas equações da cinemática aplicam-se somente a partículas com aceleração constante, sendo que não são válidas em situações com aceleração variável.

Segundo Serway (em Serway, 1996), para desenvolver-se uma estratégia geral de resolução de problemas, cinco etapas em geral devem ser utilizadas:

- a) Diagrama:** Deve-se desenvolver um diagrama do problema, com dísticos apropriados e eixos de coordenadas, se necessário.
- b) Identificação dados:** Ao ler o problema, identificar o(s) princípio(s) básico(s) envolvido(s), fazendo a listagem dos dados e das incógnitas.
- c) Equação Básica:** Escolher uma relação básica, ou deduzir-se uma equação, possível de ser usada para achar a incógnita, e resolva simbolicamente esta equação.
- d) Equação Operativa:** Substituir os valores dados pelas unidades na equação.
- e) Verificação da Solução:** Obter o valor numérico da incógnita.

O problema estará resolvido se as seguintes perguntas tiverem respostas apropriadas: *As unidades são compatíveis? A resposta é razoável? Os sinais de mais e menos são adequados e têm sentido?*

Se as respostas obtidas não forem satisfatórias o processo deverá ser reiniciado re-avaliando-se cada etapa.

Pode-se verificar o método de Serway na Figura 3.12.



Figura 3.12 - Método de Serway

Um dos objetivos desta estratégia é favorecer a exatidão. Diagramas adequados podem eliminar erros de sinal, ajudam a separar os princípios físicos do problema. As soluções simbólicas, além dos dados e incógnitas cuidadosamente classificados, ajudam a eliminar erros causados por falta de atenção. Uma verificação das unidades, no final da resolução, pode indicar um erro algébrico possível.

A disposição e a organização do problema no papel farão com que o resultado final seja mais compreensível e fácil de aprender. Uma vez que se tenha desenvolvido um sistema organizado para resolver problemas de provas e obter informações relevantes, os problemas serão abordados com muito mais confiança.

Exemplo: Uma maquinista em um trem a 20 m/s aplica os freios e pára a uma distância de 100m. Qual a aceleração do trem?

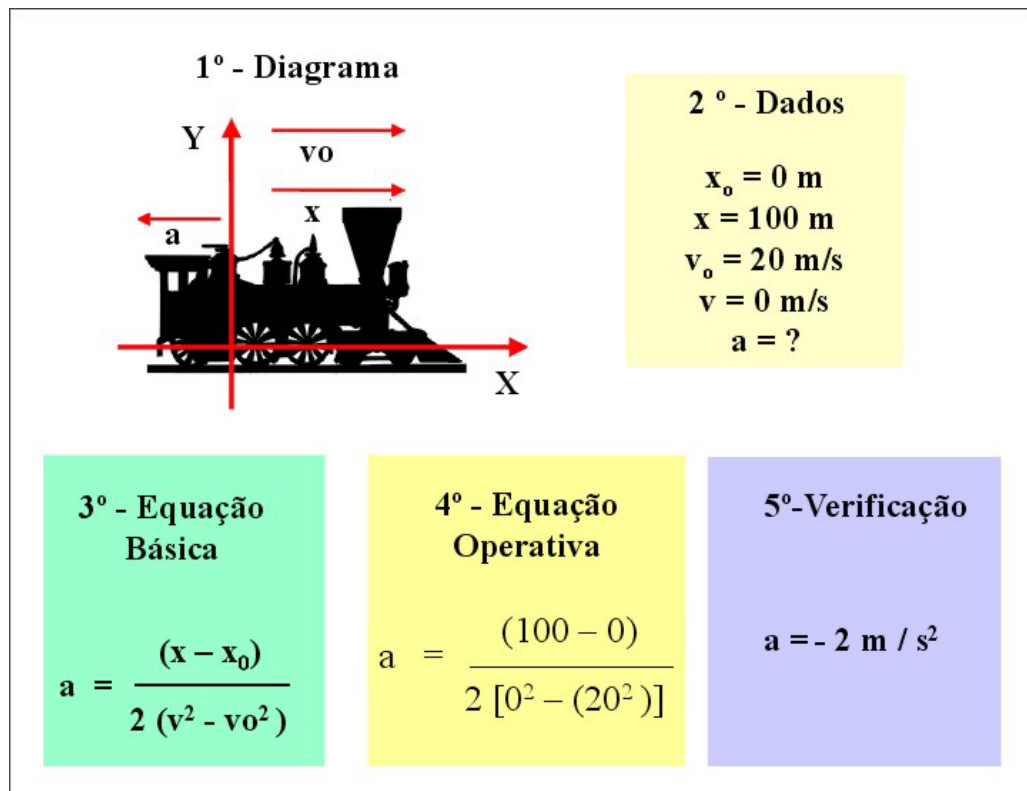


Figura 3.13 – Exemplo de Resolução de Problema pelo método do Serway

3.3.3 Mapa Mental

Os mapas mentais, também conhecidos como mapas conceituais, consistem em representações gráficas semelhantes a diagramas, que indicam relações entre conceitos. Representam uma estrutura que vai desde os conceitos mais abrangentes até os menos inclusivos.

Representando um conjunto de conceitos inter-relacionados em uma estrutura hierárquica e proposicional, os recursos esquemáticos dos mapas conceituais servem para tornar claro para professores e alunos as relações entre conceitos de um conteúdo aos quais deve ser dada maior ênfase (Novak et al., 1996).

O especialista inglês na área de desempenho mental, Tony Buzan (Buzan, 1974), desenvolveu estudos sobre o método do mapa mental, observando a necessidade de estudantes

ao redor do mundo de transpor barreiras para a compreensão e memorização de determinados assuntos, com vistas a melhorar o seu potencial de aprendizagem.

Estas barreiras são criadas em sua grande maioria por acreditar-se até então que a mente humana funciona de forma linear, com base em listas seqüenciais. Esta influência percebe-se com clareza quando nas escolas somos treinados a fazer anotações em forma de listas.

As anotações de forma geral são feitas como mostra nas figuras que seguem.

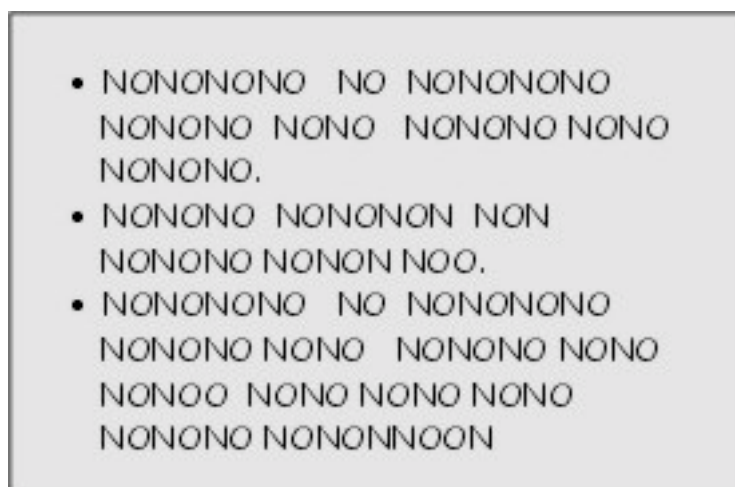


Figura 3.14 - Anotação 1

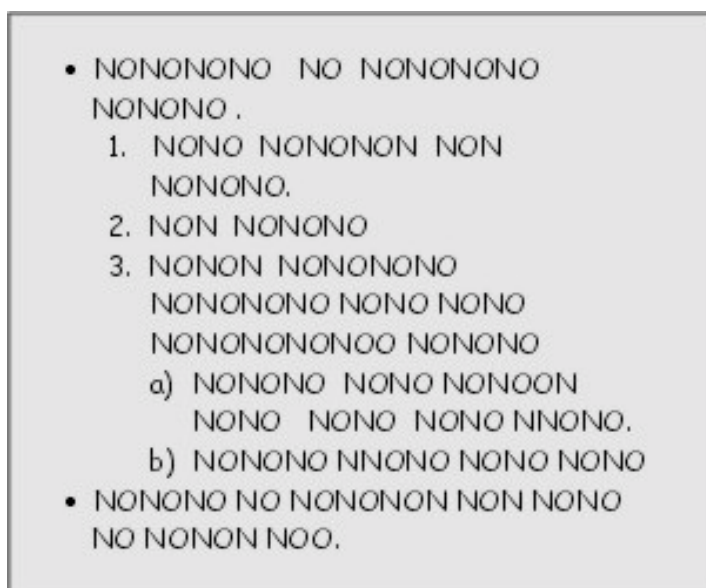


Figura 3.15 - Anotação 2

A forma como cérebro trabalha é bastante complexa, capturando inúmeras formas de comunicação, podendo ser visuais, auditivas, outras percepções não lineares como fotografias, ilustrações, etc. As informações são armazenadas e interligadas usando conceitos-chave inter-relacionados.

Outra grande dificuldade para a aprendizagem é a tendência de acreditar-se que a mente só aprende quando está funcionando racionalmente. Howard Gardner, afirma (em Gardner, 1994) que as pessoas podem ter sete tipos de inteligência:

1 – Racional: É medida com teste de QI e provas.

2 – De Padrões: Habilidade de perceber e criar padrões, comum em artistas e matemáticos.

3 – Musical: Relativa a criação de melodias, presente nos músicos.

4 – Física: Fortemente presente nos atletas.

5 – Prática: Encontrada naqueles que montam e desmontam algo sem conhecer sequer os nomes das peças.

6 – Intrapessoal: Está relacionada com o auto-conhecimento.

7 – Interpessoal: Presente em pessoas populares, que são amigas de todos.

A grande descoberta é que todos estes tipos de inteligência são portas para a aprendizagem. Portanto, com o uso da criatividade, tanto do professor quanto do aluno, pode-se explorar o potencial mental de cada aluno de forma mais eficiente.

Um mapa de uma cidade é uma representação abstrata desta cidade, sendo, portanto simplificado, muitos detalhes que para o contexto de atuação deste objeto são desnecessários.

De forma análoga, o mapa mental, quando usado para resolver problemas de um determinado assunto, deixa de lado grande parte dos detalhes inúteis focalizando a atenção nos dados essenciais do problema.

Existem quatro grandes categorias de mapas conceituais. Elas se distinguem pelas diferentes formas de representar informação e são as seguintes:

1. Mapa conceitual Aranha: é organizado colocando-se o tema central no meio do mapa e sub-temas ao redor do principal são acrescentados.

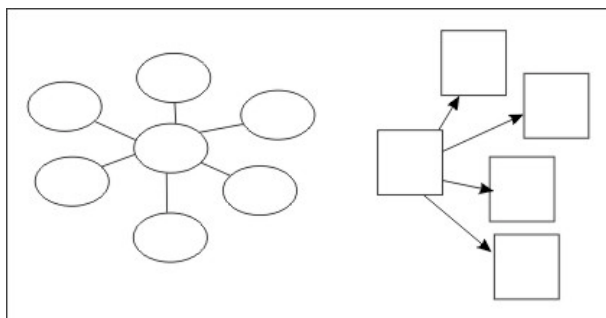


Figura 3.16 – Mapa Conceitual Aranha

2. Mapa Conceitual Hierárquico: apresenta as informações em uma ordem descendente de importância, ou seja, a informação mais importante é colocada no topo.

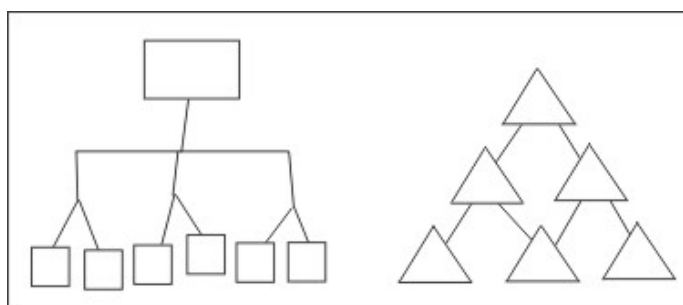


Figura 3.17 – Mapa Conceitual Hierárquico

3. Mapa Conceitual FlowChart: organiza as informações de forma linear.

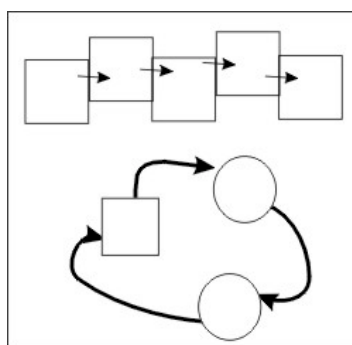


Figura 3.18– Mapa Conceitual FlowChart

4. Mapa Conceitual Sistemas: organiza as informações de forma similar ao Flowchart, apenas adicionando entradas e saídas.

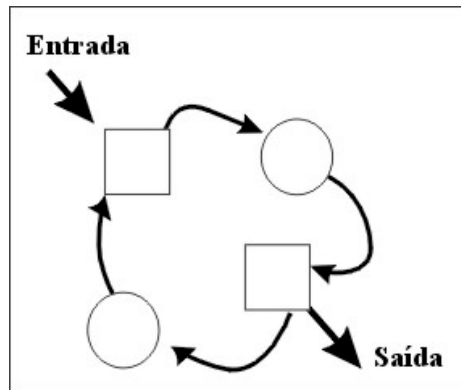


Figura 3.19 – Mapa Conceitual Aranha

O método para criação de mapas mentais consiste na construção de um desenho da realidade a ser resolvida e das circunstâncias envolvidas. A idéia básica é procurar lembrar tudo que sua mente pensa em relação à idéia principal não importando ordem ou organização.

Em primeiro lugar, ocorre a extração do objetivo do problema, sendo este destacado e a partir de onde as idéias devem surgir. Identificado-se a idéia central, passa-se então à etapa onde são relacionados conceitos-chave e às inter-relações existentes, sendo tudo isto agrupado em uma figura. No primeiro momento, como não existe preocupação com a relevância das informações, deve-se escrever o máximo conhecido a respeito do problema sem preocupação com o resultado da figura.

O próximo passo é a edição do desenho, que consiste em selecionar as partes mais importantes, juntando-as de modo a definir claramente como chegar a um bom resultado final. Podem ser usadas setas representando os relacionamentos, nuvens agrupando palavras ou conceitos e também cores para realçar ações específicas. Nesta etapa de edição do mapa, provavelmente novas associações devem surgir, podendo estas ser acrescentadas ao mapa.

Demonstrando um mapa mental, a figura que segue é um possível resultado, logicamente resumido, feito por uma pessoa que precisava escrever um currículo para apresentar a uma empresa, a qual usando da técnica, conseguiu descrever o máximo de informações relevantes ao seu propósito.

Especificando apropriadamente a partir do desenho, somente os dados úteis, sem se perder em informações desnecessárias, o que possivelmente acarretará um prejuízo.

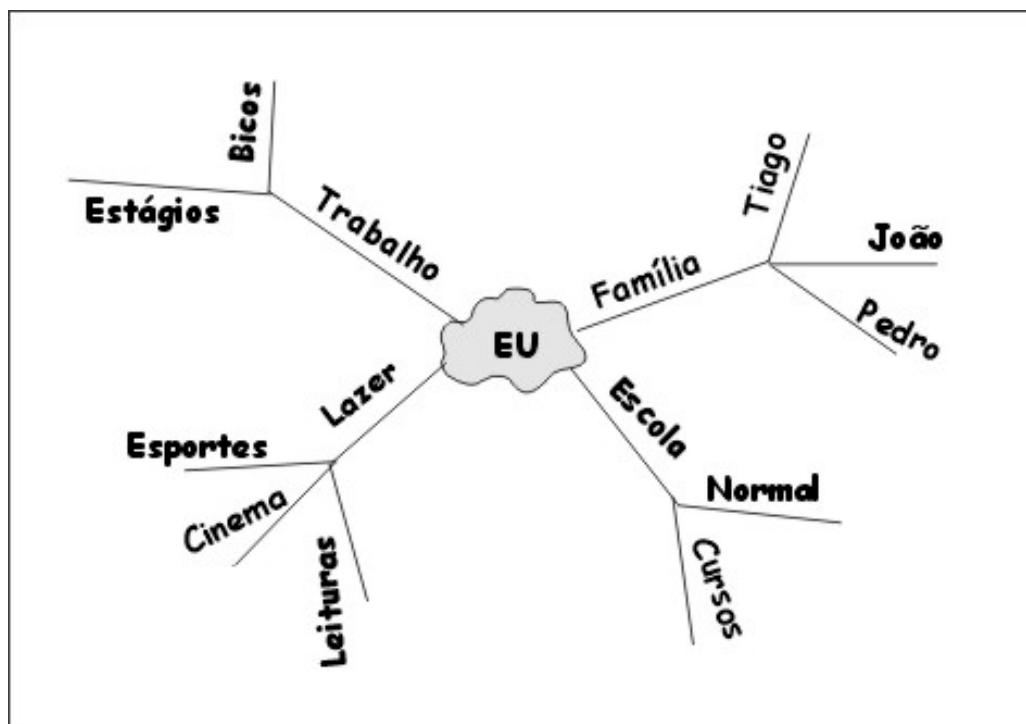


Figura 3.20 – Exemplo Mapa Mental

Através de suas pesquisas, Tony Buzan, comprovou que as anotações feitas pelos alunos na forma tradicional, ou seja, baseado nas listas de sentenças, 90% das palavras utilizadas são desnecessárias para efeito de memorização e conseqüentemente aprendizado. Esta descoberta é apoiada nos seguintes fatos:

- a) O estudante desperdiça tempo anotando informações que não influem na memorização;
- b) O aluno desperdiça tempo relendo as palavras desnecessárias;
- c) Palavras-chave não são distinguidas por nenhuma marcação;
- d) As conexões entre as palavras-chave existentes no texto são interrompidas por palavras desnecessárias, denominadas de ruídos. E como o cérebro trabalha com associações, estas interferências atrapalham as conexões.

A técnica do mapa conceitual ajuda o estudante a evitar todos estes desperdícios, pois estimula o processo de associação com base em palavras-chave. No mapa mental existe lugar para todos os conceitos e idéias que aparecem. Além disto, um mapa mental é muito mais fácil de ser decifrado, após algum tempo, do que anotações tradicionais. O padrão visual, apoiado por cores, figuras e setas torna mais fácil para o cérebro retornar à situação em que o mapa foi criado. Desta forma, as oportunidades de aprendizagem levam os alunos a adquirirem um conhecimento permanente.

A técnica de mapa mental não nos conduz a uma solução, mas nos conduz a uma clara organização de idéias, resultando em uma captura de dados eficiente e estabelecimento de conexões eficazes.

A Figura 3.21 mostra o resultado de um mapa mental desenhado a partir do seguinte enunciado de um problema proposto comum: *A que velocidade estaria um automóvel que percorre 100 km em 2h?*

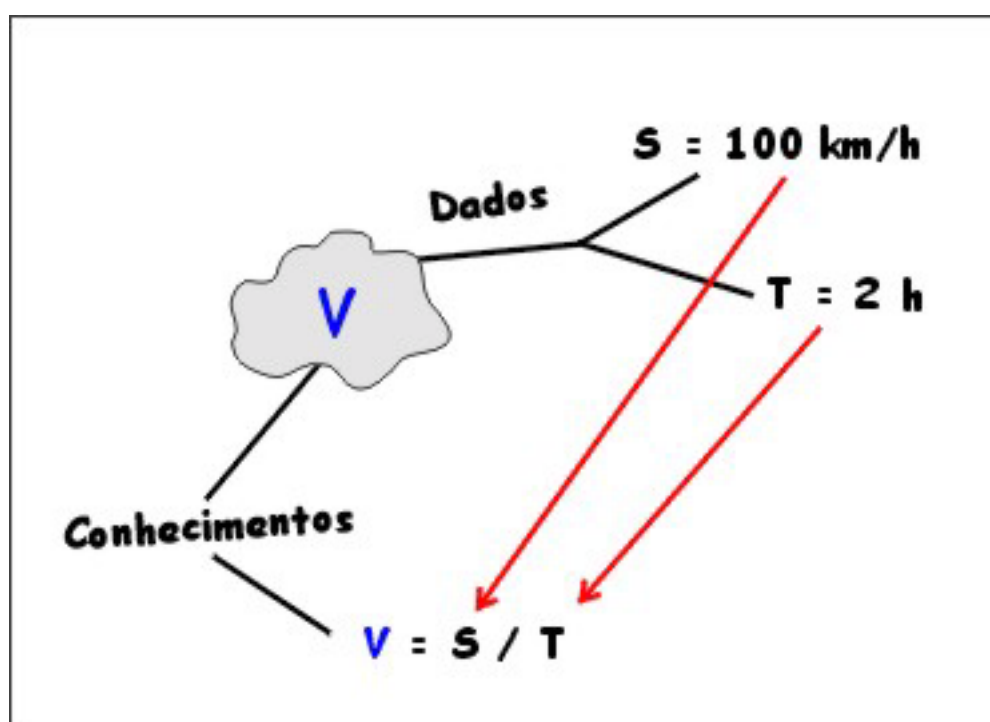


Figura 3.21 – Mapa Mental Problema

3.4 Considerações Finais

Este capítulo teve como objetivo fornecer uma visão geral do que são problemas, e devido à grande abrangência do seu significado, definir com que tipo específico de problemas estaremos trabalhando e ainda analisarmos alguns métodos de resolução de problemas que servirão como base na definição do funcionamento do Assistente de Resolução de Problemas do Sistema Mathnet.

Como já foi apresentado o ambiente da nossa proposta no Capítulo 2 e alguns fundamentos sobre resolução de problemas no Capítulo 3. No capítulo que segue estaremos unindo estes dois pontos, ou seja, especificaremos como resolver problemas no Projeto Mathnet.

4. RESOLUÇÃO DE PROBLEMAS NO MATHNET

Considerando o seu caráter pedagógico, o Sistema MATHNET necessitava de uma ferramenta que auxiliasse na avaliação geral, levando em consideração o grau de aprendizagem e fixação dos assuntos ministrados aos estudantes e, até mesmo, que pudesse quantificar através de avaliações (provas) esta aprendizagem, e desta forma questionando a eficiência apresentada pelo sistema.

Esta necessidade implica, portanto, ao aperfeiçoamento de toda estrutura do sistema no que diz respeito à forma como os conteúdos são ministrados e as atividades de grupo e individuais são realizadas.

Conforme o resultado geral desta avaliação dos alunos, se for negativo, poderá ser feita uma avaliação no sistema de ensino adotado para detectar os métodos que precisam ser reestruturados devido a sua baixa performance apresentada, ou inadequações ao público alvo. Fazendo isto, é acrescido ao sistema como um todo a possibilidade de maximizar seus resultados. Quando o resultado for positivo, verifica-se a existência de possibilidades para aperfeiçoamento, otimizando as etapas que já estão bem definidas.

O Sistema MATHNET, como foi descrito anteriormente, funciona através da interação dos alunos, dos grupos, dos professores e dos agentes. O Assistente de Resolução de Problemas utiliza a funcionalidade destas interações, em momentos específicos quando requisitado, tendo como objetivo a modelagem, o armazenamento e a resolução de problemas propostos e também análise de uma dada solução.

4.1 Atividades Pedagógicas

O Assistente de Resolução de Problemas pode ser utilizado em três fases da sequência de atividades pedagógicas definidas no sistema MATHNET.

Nas diferentes fases do Sistema Pedagógico que utilizam o Assistente de Resolução de Problemas, todas as etapas para a resolução são utilizadas igualmente, diferindo entre elas o enfoque ou objetivo requerido.

Abaixo, especificaremos como o Assistente de Resolução de Problemas irá interagir em cada uma das três fases que estão destacadas na figura seguinte de verde.

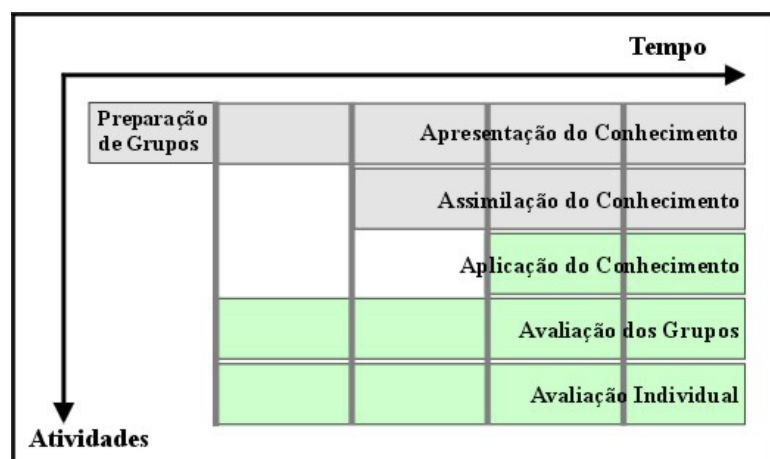


Figura 4.1 – Atividades pedagógicas que podem utilizar o Assistente

- a) Aplicação do Conhecimento:** nesta fase, o conhecimento já supostamente assimilado, é apresentado ao grupo em forma de tarefas para que o mesmo possa resolvê-las de maneira interativa com os agentes. Neste ponto, as tarefas são atividades práticas que pretendem levar o aprendiz a uma aproximação do contexto ministrado e uma melhor assimilação e fixação.

Tanto a fase de Avaliação do Grupo como a de Avaliação individual pode ocorrer paralelamente à fase de Aplicação do Conhecimento, sendo que as tarefas propostas servem aos dois propósitos.

- b) Avaliação do Grupo:** esta atividade pedagógica avalia o grupo, sendo o enfoque aqui, quantificar o tanto que o conjunto de alunos, com a cooperação e integração de cada membro, reteve do conhecimento.
- c) Avaliação Individual:** o aluno é avaliado individualmente, levando em consideração os resultados obtidos na resolução de seus problemas e até mesmo o tempo utilizado para responder. Nesta fase, cada aprendiz tem definido seu

aproveitamento e também em que assuntos se faz necessário reforço. Neste processo, não é permitida a cooperação do aprendiz com outros alunos ou agentes.

4.2 Metodologia

A partir da abstração dos métodos de resolução de problemas do Pólya (Pólya, 1945), e do Serway (Serway, 1996), definimos um modelo para o aprendiz efetivar o processo de resolução de problemas, onde foram determinadas seis etapas bem específicas, visando com isto a uma melhor interação tanto para o professor na definição e especificações para os problemas, quanto para o aprendiz na resolução do mesmo.

A Figura 4.2 mostra o relacionamento implícito entre o método do George Pólya e do Serway, sem levar em consideração o grau de abrangência de cada ponto, apenas considerando a essência de cada um.

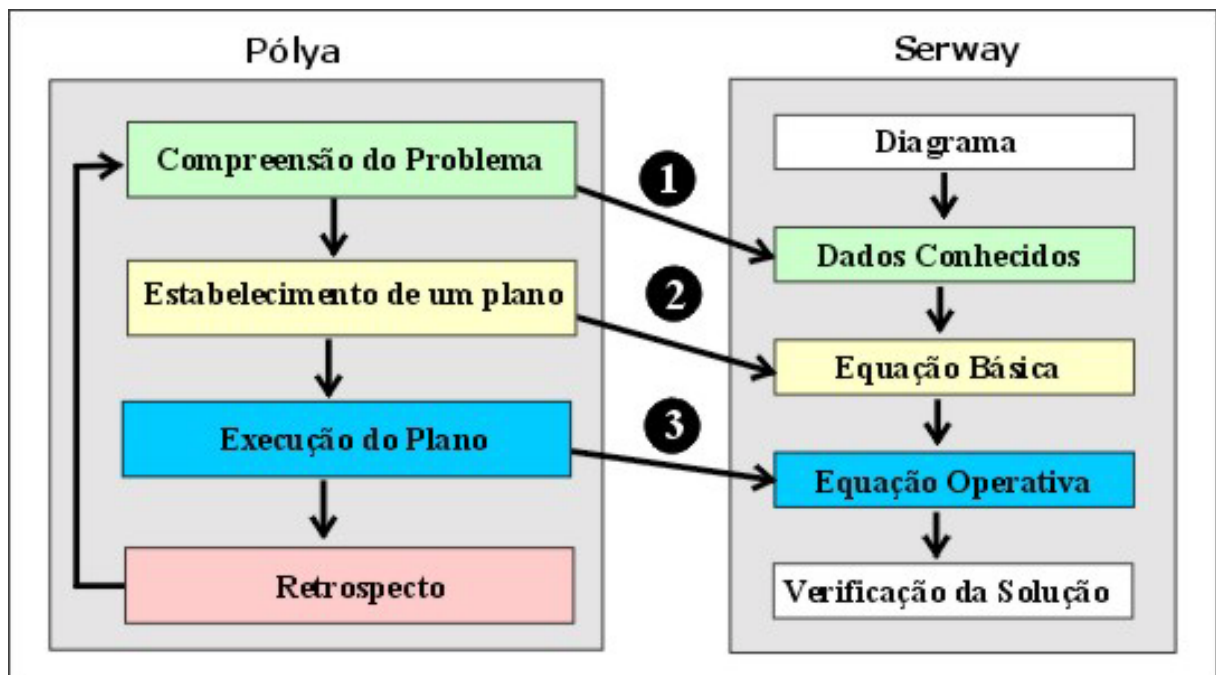


Figura 4.2 – Relacionamento entre o método de Pólya e do Serway.

As etapas do modelo definido para o aluno resolver problemas no Mathnet são as seguintes:

1. Identificação dos objetivos: O aprendiz deverá, a partir de um enunciado, identificar qual é a incógnita principal do problema, ou seja, reconhecer o que o problema está pedindo. Esta etapa definirá em função de quem o resultado será apresentado e, logicamente, uma especificação incorreta do objetivo, mesmo com a solução parcialmente correta, acarretará em prejuízo ao aprendiz.

2. Coleta de Dados: Também, a partir do enunciado, o aluno deverá reconhecer os dados iniciais presentes, explícitos, que servirão como base para resolução do problema. Estes dados fornecidos no enunciado serão, certamente, utilizados posteriormente, o que faz necessário sua atenta observação.

3. Verificação de Conhecimentos Necessários: Está relacionado com os conhecimentos relativos ao domínio do problema proposto, os quais o aprendiz já deverá possuir e saber tanto identificá-los quanto manipulá-los, em consequência de esta etapa ser anterior ao processo pedagógico. Os problemas estão necessariamente relacionados com o assunto ministrado pelo professor; logo, os conceitos e fórmulas já foram previamente esclarecidos.

Utilizam-se as idéias representadas nas etapas destacadas pelo relacionamento 1 na Figura 4.2, descritas pelo Pólya e Serway, devido a suas semelhanças intrínsecas, para avaliar a compreensão geral do aluno em relação ao problema no que diz respeito aos dados iniciais provenientes do enunciado, os conhecimentos necessários e também à identificação do objetivo.

4. Estratégia: Elaboração de um caminho lógico a ser percorrido, especificação de passos, a partir dos conhecimentos iniciais (analisados na fase de coleta de dados e verificação de conhecimentos), que levam a uma possível solução. A estratégia adotada é pessoal a cada aluno, podendo ser totalmente distinta de um aprendiz para outro. Os passos de resolução correspondem à maneira como o estudante organiza as informações em seu poder, para encontrar uma informação que não possui.

Na definição da estratégia, ou simplesmente dos passos para resolução do problema, no MATHNET utilizaram-se os conceitos das etapas do relacionamento 2 da Figura 4.2. No

caso do Pólya, é bastante abrangente em sua especificação, possuindo por isso uma grande quantidade de informações relevantes, e no caso do Serway é mais simplificada, contendo apenas o básico para resolução.

5. Execução: Aplicar todos os passos definidos na etapa anterior para encontrar uma solução para o problema. A atenção durante a etapa de execução é extremamente importante para evitar-se equívocos como troca de valores ou sinais, o que fatalmente conduzirá a um resultado final errado. A execução realiza as substituições, manipulações algébricas e atribuições de valores.

A fase de execução ou da aplicação da estratégia adotada, que é de fundamental importância para a obtenção de um resultado correto e que representa a validação de todo esforço até aqui, é baseada nos conceitos contidos nas etapas do relacionamento 3 na Figura 4.2.

6. Verificação: Todo o processo de resolução é revisto, em busca de uma possível falha, ou até mesmo da possibilidade de melhorar a solução. Deve, certamente, existir mais de uma maneira de resolver um problema. Portanto, esta fase é destinada à observação, se realmente tudo que foi especificado está coerente e da possibilidade de, por exemplo, diminuir a quantidade de passos para chegar-se a solução.

Finalmente, os princípios da etapa de Retrospecto do Polya serão aplicados na verificação da veracidade do resultado obtido, levando em consideração que deve existir um alto grau de confiabilidades nos resultados apresentados.

A técnica de Mapa Mental poderá ser utilizada, visando-se a gerar uma representação gráfica de todas as etapas do processo até sua resolução, para melhor fixação e aprendizagem do aprendiz e entendimentos de outra pessoa que venha consultá-lo.

Estas etapas, não necessariamente, precisam ocorrer de forma sequencial, uma exatamente após a outra. Na verdade, o processo se dá concorrentemente em vista da necessidade de validação a cada etapa ou operação realizada. Sendo, portanto, um tanto quanto difícil a localização ou especificação onde exatamente está ocorrendo cada etapa.

A Figura 4.3, mostra as etapas do modelo definido para o aprendiz resolver problemas no Mathnet.

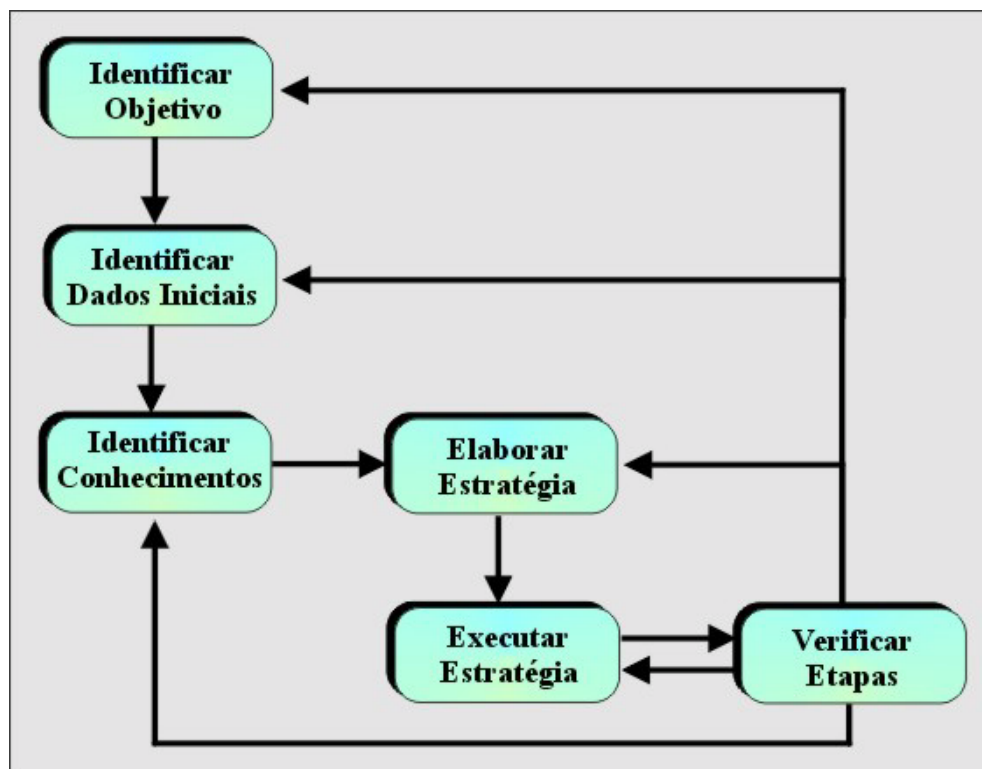


Figura 4.3 – Metodologia de Resolução de Problemas do Mathnet.

4.3 Arquitetura Mathnet

A proposta do Assistente de Resolução de Problemas para o Mathnet, conforme a Figura 4.3 consiste basicamente em:

1. Uma interface de cadastro de problemas: Na qual o professor, após ministrar seu conteúdo, cadastrará problemas referentes ao seu domínio, para fixação e/ou avaliação do aprendiz ou grupos, levando em consideração três níveis de dificuldade (básico, médio e avançado) em vistas do adequamento ao perfil dos alunos.

2. **Uma interface de resolução de problemas:** Onde o aluno efetivamente solicitará e resolverá seus problemas e que é composta por um conjunto de outras interfaces, sendo cada uma destinada a um fim específico no processo de resolver problemas definido anteriormente.

3) **Uma classe problema:** por onde são modelados os problemas cadastrados pelo professor e utilizada para resolver problemas.

4) **Uma classe solução:** que é referente às soluções produzidas pelos aprendizes ou pelo próprio assistente.

5) **Funcionalidades agregadas ao agente tutor:** que constituem o núcleo do processo de resolver e analisar problemas.

A este conjunto de objetos, chamamos Assistente de Resolução de Problemas do Sistema Mathnet, e pode ser visto na Figura 4.4.

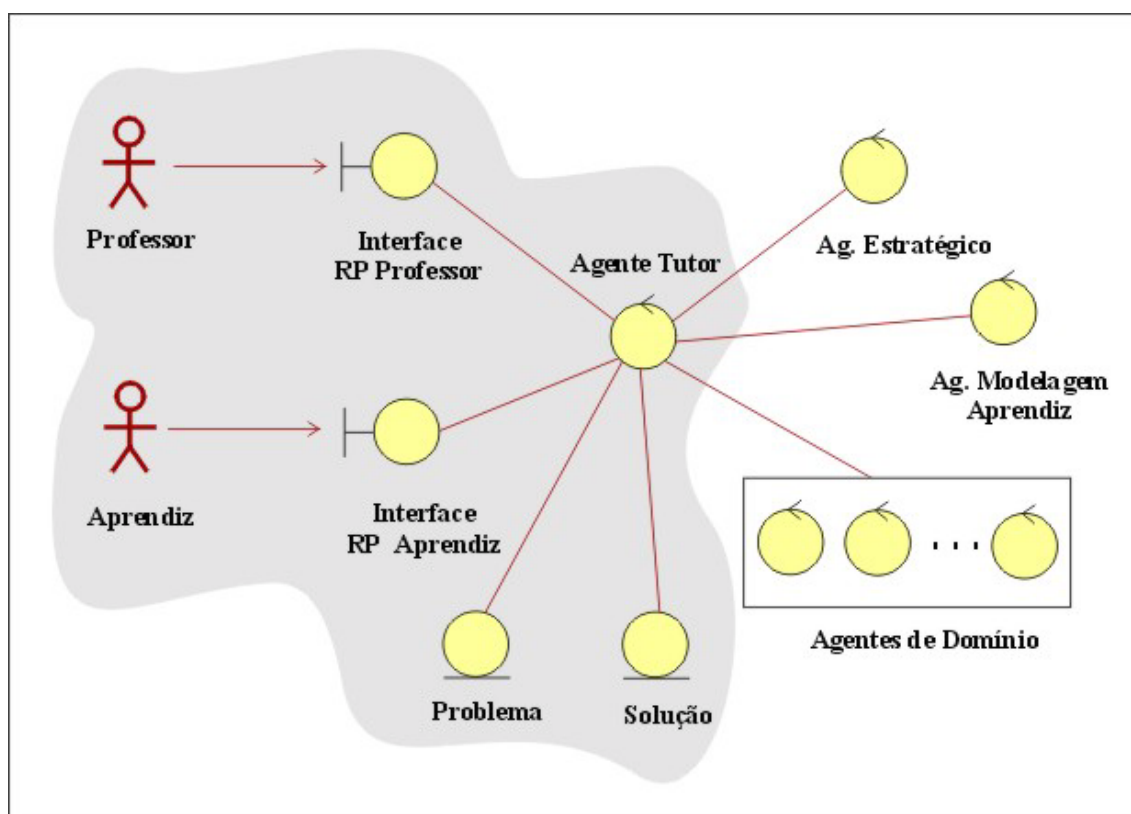


Figura 4.4 – Proposta Arquitetura Mathnet.

4.4 Modelagem do Processo de Resolução de Problemas

A modelagem no projeto Mathnet é realizada usando Unified Modeling Language - UML (Júnior, 2000). Apresentamos aqui os casos de uso envolvidos no processo de resolução de problema: do Professor e do Aprendiz, deixando para a próxima seção o detalhamento do caso de uso resolver problemas.

4.4.1 Caso de Uso Professor

O agente humano, professor, responsável por um dado domínio, é incumbido da tarefa de cadastrar problemas de fixação, levando em consideração os assuntos ministrados.

Através de uma interface, os problemas são cadastrados juntamente com as especificações requeridas pelo sistema para sua resolução e armazenados em uma base apropriada.

A Figura 4.5 mostra o caso de uso do professor.

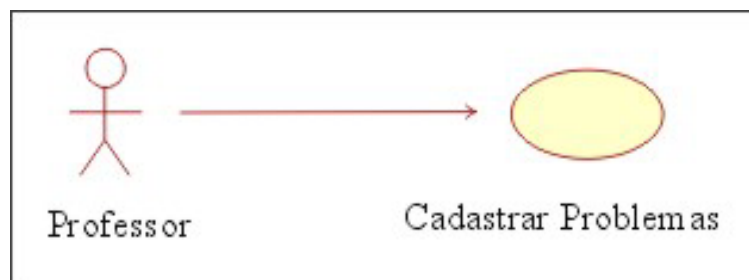


Figura 4.5 – Diagrama do caso de uso Professor

4.4.2 Caso de Uso Aprendiz

O Aluno também possui uma interface específica onde poderá solicitar exercícios de uma determinada disciplina para resolver, ou então ser enviado a ele uma avaliação de conteúdo.

Uma lista de problemas, relacionada com o assunto especificado, é retornada ao aprendiz com o resumo de cada um. Sendo necessário neste momento a escolha de um dos problemas para resolver e desta maneira são disponibilizadas todas as informações disponíveis referentes ao problema.

O problema é livremente trabalhado pelo aluno, ou com restrição de tempo no caso de avaliações.

Quando o aluno informa que chegou ao final da resolução, a solução especificada é armazenada, podendo ser emitido ou não o resultado da verificação conforme a situação.

O caso de uso aprendiz é demonstrado na Figura 4.6.

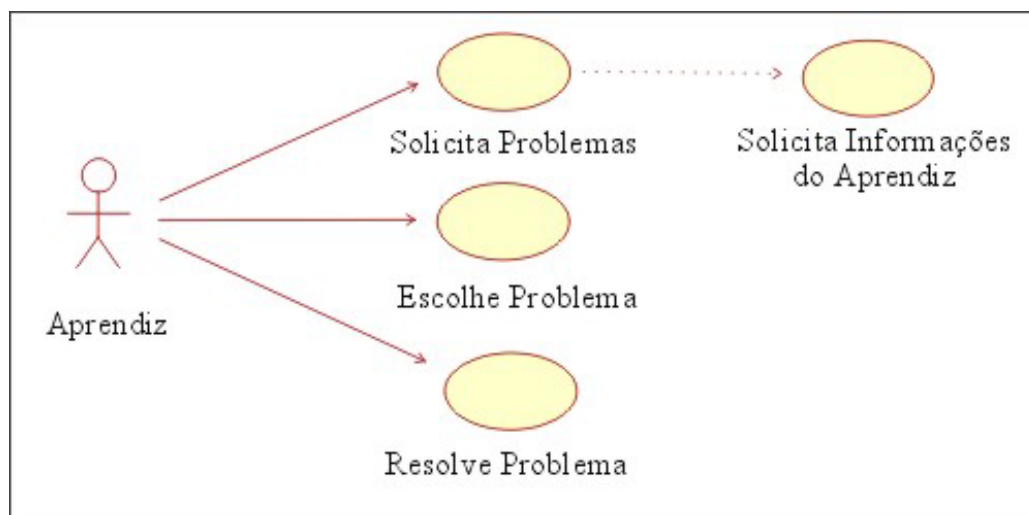


Figura 4.6 – Diagrama do Caso de uso Aprendiz

Este procedimento pode ser repetido até que todos os problemas sejam resolvidos ou encerrados a critério do aprendiz.

Quando um aluno informa que quer resolver problemas, o Assistente de resolução solicita que o Agente Tutor verifique o perfil do usuário para, então, selecionar na base de problemas, os que pertencem ao domínio solicitado pelo aprendiz.

Baseado nas informações contidas no perfil do aprendiz, que foram informadas pelo agente tutor, o assistente filtrará os problemas encontrados, gerando uma lista de problemas em conformidade com o perfil apresentado pelo aluno.

Esta filtragem também possibilita a verificação do progresso de cada aluno, em vista do aumento do grau de dificuldade dos problemas selecionados.

4.5 Caso de Uso Resolver Problema

A tarefa de o aluno resolver problemas seguirá os fluxos de funcionamento descritos a seguir.

4.5.1 Fluxo Principal da Resolução de Problemas

1. O Assistente cria uma lista de problemas, baseado no perfil do aluno e exibe para o Aprendiz.
2. O aluno seleciona um problema para resolver.
3. O Assistente exibe os detalhes do problema, tais como: enunciado completo, figuras, etc., juntamente com ferramentas para sua resolução.

4. O aluno resolve o problema utilizando um ou mais dos seguintes passos:
 - a. Identificar o objetivo do problema.
 - b. Identificar os dados do problema.
 - c. Identificar os conhecimentos necessários.
 - d. Elaborar uma estratégia adequada
 - e. Aplicar os passos acima pra resolver o problema.
5. O Aprendiz avisa o sistema que acabou.
6. O Assistente analisa a solução e exhibe ao aluno um resumo.
7. Os passos 2 a 6 se repetem até acabar a lista de problemas.
8. O caso de uso termina com sucesso.

4.5.2 Fluxos Alternativos da Resolução de Problemas

- Fluxo 1a: Não existe lista de problemas para o aluno.
 - 1a1. O Assistente avisa ao aluno que não existe lista de problemas.
 - 1a2. O Assistente informa ao professor que deve criar os problemas para a Banco de Problemas.
 - 1a3. Caso de uso termina sem sucesso.
- Fluxo 2a : O Aprendiz não seleciona um problema para resolver.
 - 2a1. O Assistente avisa ao aluno desta necessidade.
 - 2a2. O fluxo continua do passo 3.

- Fluxo 3a : Detalhes do problema e ferramentas não disponíveis.
 - 3a1. O Assistente informa ao professor o ocorrido.
 - 3a2. O Assistente informa ao aluno de que não poderá continuar.
 - 3a3. Caso de uso termina sem sucesso.
- Fluxo 4a : O Aluno não consegue resolver o problema.
 - 4a1. O Aluno pede ajuda ao sistema.
 - 4a2. O Assistente identifica até que fase o Aprendiz chegou e sugere um próximo passo.
 - 4a3. O fluxo retorna ao passo 4.
- Fluxo 5a : O Aprendiz não informa o término do problema.
 - 5a1. O Assistente informa ao Aprendiz que é necessário encerrar o problema.
 - 5a2. O fluxo continua do passo 6.
- Fluxo 6a : O Assistente não consegue avaliar a solução.
 - 6a1. O Assistente avisa ao aluno o ocorrido.
 - 6a2. O fluxo continua do passo 7.
- Fluxo 7a : A lista de problemas não é totalmente resolvida.
 - 7a1. O Assistente informa ao Aprendiz que é necessário resolver todos os problemas.
 - 7a2. O fluxo continua do passo 2.

4.6 Considerações Finais

Retratamos neste capítulo como o Sistema Mathnet funcionará com a implantação do Assistente de Resolução de Problemas, especificando em quais atividades pedagógicas pode ser utilizada uma metodologia para resolver problemas e a modelagem do processo de como a resolução de problemas efetivamente acontecerá no ambiente Mathnet.

Na sequência, veremos as especificações e comportamentos do Assistente de Resolução de Problemas, sua análise de requisitos, estrutura e como funcionam os processos de resolução de problemas e análise de soluções.

5. ASSISTENTE DE RESOLUÇÃO DE PROBLEMAS

Este capítulo apresentará as especificações do Assistente de Resolução de Problemas do Sistema Mathnet, que poderá ser referenciado a partir deste momento apenas por ARP.

5.1 Análise de Requisitos

A análise de requisitos do ARP foi baseada nos princípios da Engenharia de Software (Wilson, 2001), que tem demonstrado na Figura 5.1 uma visão geral de suas atividades.

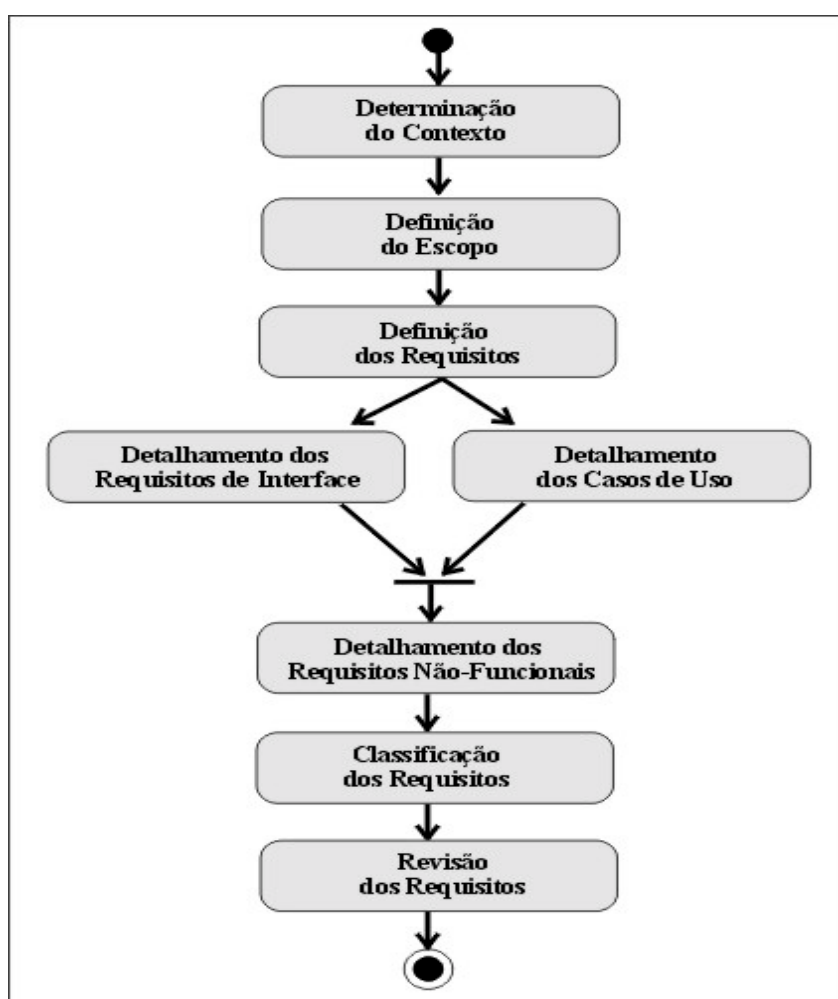


Figura 5.1 – Atividades da Análise de Requisitos

O fluxo de requisitos é composto por atividades que visam a obtenção clara e precisa das necessidades de um software, identificando desta forma, requisitos de alta qualidade, completos, sem ambigüidade, implementáveis, consistentes e testáveis.

O fluxo de requisitos consiste, portanto, nas seguintes etapas:

- ***Determinação do Contexto:*** Aqui, são levantados os aspectos dos processos de negócio ou de um sistema maior que sejam determinantes para determinação dos requisitos do produto.
- ***Definição do Escopo :*** Delimitam-se os problemas que o produto se propõe a resolver.
- ***Definição dos Requisitos:*** Produz-se uma lista de requisitos funcionais e não-funcionais escritos de forma sucinta. São também identificados os grupos de usuários do produto e as demais restrições.
- ***Detalhamentos dos Requisitos de Interface:*** São levantados os aspectos das interfaces do produto que os usuários consideram requisitos.
- ***Detalhamentos dos Requisitos Funcionais:*** Utiliza a notação de casos de uso, sendo que cada casa de uso representa uma funcionalidade do produto.
- ***Detalhamentos dos Requisitos Não-Funcionais:*** Complementação dos requisitos descreve os requisitos de desempenho e outros aspectos necessários para que o produto atinja a qualidade desejada.
- ***Classificação dos Requisitos:*** Determina as prioridades relativas dos requisitos e avalia a estabilidade e a complexidade da realização.
- ***Revisão dos Requisitos:*** Determinam-se se todos eles satisfazem os critérios de qualidade dos requisitos e se a Especificação de Requisitos está clara e bem entendida.

5.1.1 Determinação do Contexto

O desenvolvimento do Assistente se tornou necessário devido ao sistema Mathnet precisar de um mecanismo para auxiliar a avaliação dos aprendizes e grupos; portanto, em meios às funcionalidades já desenvolvidas no Sistema, se encontra o Assistente de Resolução de Problemas que, de forma resumida, tem seu contexto apresentada na Figura 5.2.

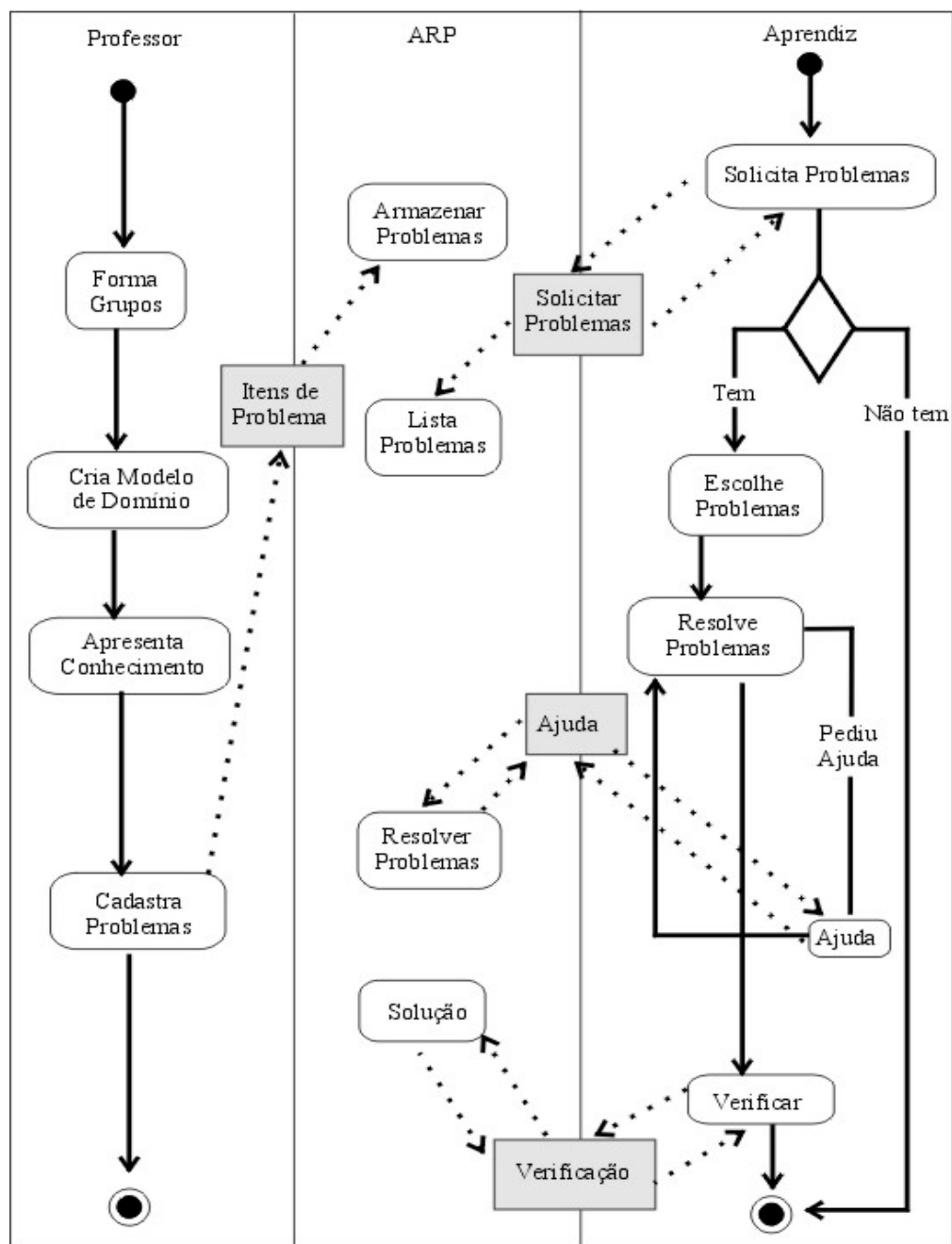


Figura 5.2 – Contexto do Assistente

5.1.2 Determinação do Escopo

a) Missão: O ponto focal de um produto é a sua missão, Tabela 5.1. A declaração da missão descreve de forma concisa o objetivo do produto, delimitando suas responsabilidades e especificando os limites e benefícios.

O Assistente de Resolução de Problemas visa a oferecer um mecanismo de resolução de problemas e outro de verificação de soluções para o Sistema Mathnet.

Tabela 5.1 - Missão

b) Limites: O Assistente de Resolução de Problemas tem as suas restrições devido ao alto grau de complexidade neste tipo de implementação. Estas restrições, ou aquilo que o sistema não pode fazer, estão especificadas na tabela que segue.

- O Assistente trabalhará apenas com equações simples e suas combinações primárias, que são dos seguintes tipos:

$a = b + c$	$a = b - c$
$a = b * c$	$a = b / c$

- Cada variável deverá ser representada por um único literal, ou seja, V é aceito, mas Vo não.
- Os problemas poderão ter apenas um objetivo.
- No caso da necessidade do uso de estratégias, as mesmas devem ser especificadas em conjunto com os conhecimentos iniciais.
- O Assistente não se responsabiliza pelas atividades de backup dos problemas e soluções do aluno, ficando a cargo dos administradores.

Tabela 5.2 - Limites

c) Benefícios: Os benefícios esperados como resultado do Assistente são especificados na Tabela 5.3, bem como seu grau de importância para o Sistema Mathnet. O levantamento de benefícios é importante para fazer o comparativo se os resultados obtidos compensam os recursos empregados.

Número de ordem	Benefício	Importância
1	Cadastro de problemas	Essencial
2	Avaliação do Aprendiz	Essencial
3	Avaliação dos Grupos	Essencial
4	Resolução de Problemas	Essencial
5	Verificação de Soluções	Essencial
6	Ajuda ao Aprendiz na Resolução de Problemas	Essencial
7	Diminuição das tarefas do Professor	Desejável

Tabela 5.3 - Benefícios

5.1.3 Definição dos Requisitos

Nesta atividade são identificados os casos de uso, que são representações de funções do produto, e os atores, que são representações dos usuários e de outros sistemas que interagem com o produto.

a) Casos de Uso: Os casos de uso representam funções completas do ARP e estão representados na Figura 5.3 e os seus resumos estão descritos sucintamente na Tabela 5.4.



Figura 5.3 – Casos de Uso

Número de ordem	Caso de Uso	Descrição
1	Cadastro Problemas	O professor cadastra problemas de uma disciplina ministrada, especificando os dados iniciais, conhecimentos e objetivo.
2	Lista Problemas	O Assistente, baseado em informações do Agente Tutor sobre o perfil do aprendiz, carrega, dentre todos os problemas cadastrados, os compatíveis com o aluno que solicitou.
3	Resolução de Problema	Dado um problema, o Assistente gera uma solução.
4	Verificação de Solução	Dado uma solução, o Assistente checa a veracidade das informações e se com as etapas especificadas é possível chegar ao resultado correto.

Tabela 5.4 – Casos de uso

b) Atores: Os papéis dos usuários do Assistente de Resolução de Problemas são modelados através dos atores conforme a Figura 5.4, tendo uma breve descrição na Tabela 5.5. Os atores modelam os papéis e não as pessoas dos usuários, ou seja, o mesmo usuário físico pode agir como mais de um tipo de ator.

Os agentes internos ou externos (outros sistemas, hardwares, Tutor, Estrategista, etc.) também podem ser modelados como atores.



Figura 5.4 – Atores

Número de ordem	Ator	Descrição
1	Professor	Agente Humano, responsável por cadastrar os problemas.
2	Aprendiz	Agente Humano, que irá solicitar os problemas, resolvê-los podendo tirar suas dúvidas e solicitar na'lise de soluções..
3	ARP	Agente Artificial, que pode gerar uma solução para um problema e também checar uma solução.

Tabela 5.5 - Atores

c) Relacionamento entre os Casos de Uso e Atores: Os diagramas de casos de uso especificam os relacionamentos entre casos de usos e os atores, como podem ser vistos nas Figuras 5.5, 5.6 e 5.7. Os relacionamentos indicam a existência de comunicação entre os atores e casos de uso.

Um caso de uso pode estar associado a mais de um ator, quando sua execução requerer isto.

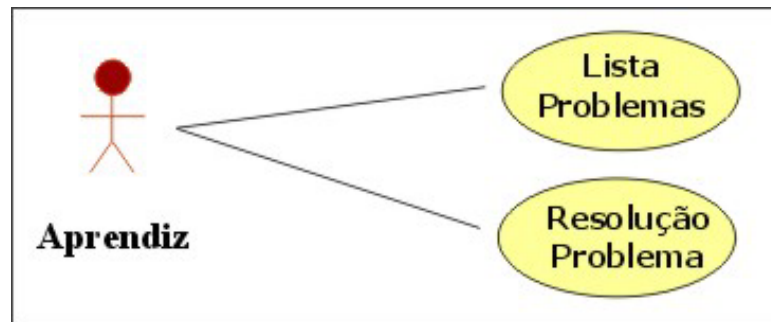


Figura 5.5 – Relacionamento Aprendiz – Lista de problemas

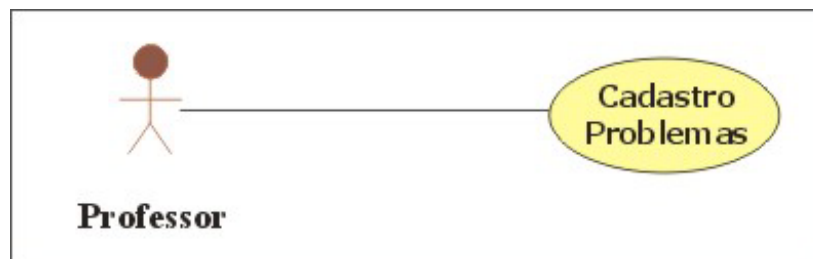


Figura 5.6 - Relacionamento Professor – Cadastro de Problemas

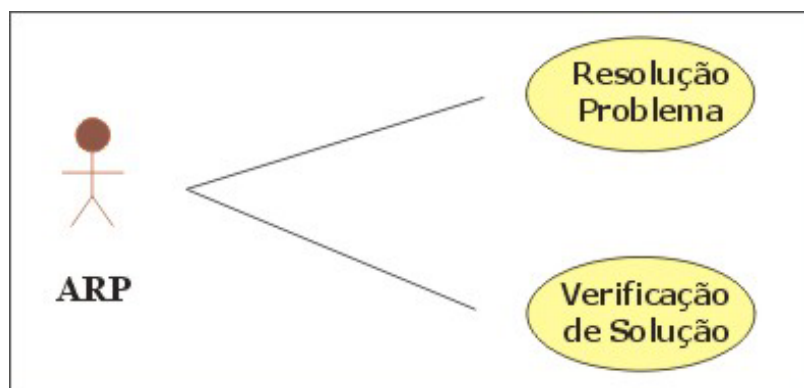


Figura 5.7 - Relacionamentos ARP

d) Diagrama de Contexto: O diagrama de contexto é o diagrama de casos de uso mais importante e, neste caso de uso, os usuários, sistemas externos e outros componentes, são representados por atores, enquanto os casos de uso representam as possíveis formas de interação do produto com os atores.

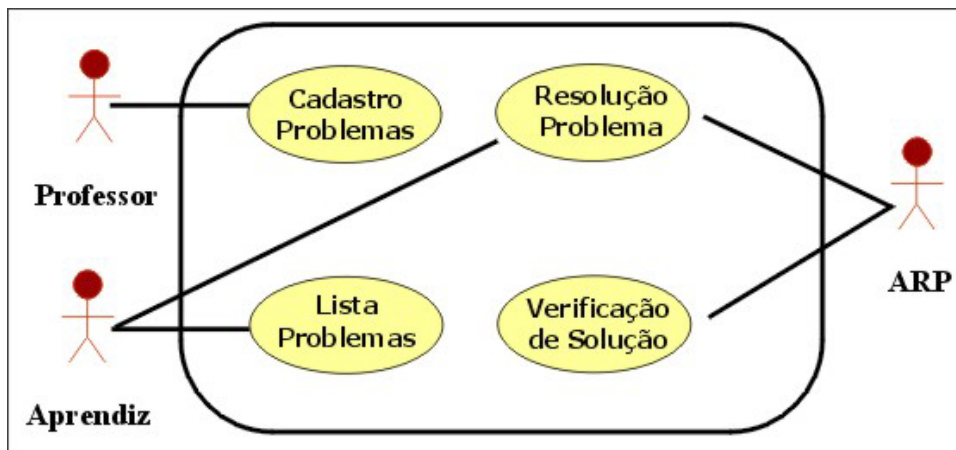


Figura 5.8 – Diagrama de Contexto

5.1.4 Detalhamento dos Requisitos de Interface

a) Interfaces Genéricas: Nesta atividade foram levantados os requisitos referentes às entradas e saídas do Assistente de Resolução de Problemas, incluindo os tipos de dados compartilhados com outros componentes do Sistema Mathnet.

1. Interface da Resolução de Problemas

1.1 Fonte de Entrada

Arquivo em XML (Júnior, 2000)

1.2 Destino de Saída

Arquivo em XML

1.3 Relacionamento com outras interfaces

Lista de Problema gera lançamento para esta interface

Tabela 5.6 – Requisitos para Interface do Assistente

b) Interfaces Gráficas de Usuário: Nas Interfaces Gráficas de Usuário, existem questões que representam requisitos do produto, tais como: formato dos dados e comandos. Os campos e comandos incluídos em cada interface de usuário, representam requisitos de captura e exibição de informações.

As figuras que seguem mostram as principais interfaces do Assistente de Resolução de Problemas, sendo demonstrado todas as etapas usadas no processo geral de resolução de análise de problemas.

A interface Cadastro de Problemas permitirá ao professor cadastrar os problemas do seu domínio. Os dados informados serão criticados, verificando-se com isto sua validade e, então, é gerado um arquivo, utilizando-se as regras de XML, onde o problema ficará armazenado para futuras utilizações.

ARP - Cadastro de Problemas

Disciplina Física

Título Problema 01

Nível
☐ Básico ☐ Médio ☐ Avançado

Enunciado Completo
 Quantos kilometros um carro percorre estando com velocidade de 20 km/h se ele andou durante 3h e depois mais 7h com a mesma velocidade?

Enunciado Resumido Deslocamento

Objetivo
Descrição Espaço **Variável** S

Dados

Descrição	Variável	Valor	Unidade
Velocidade	V	= 20	Km/h

Conhecimentos
Descrição $V = S / T$

Listagem do Problema
 Tempo Inicial : A = 3 h

Limpar Salvar

Figura 5.9 – Interface de Cadastro de Problemas

A primeira interação do aluno com resolução de problemas ocorre através da interface que segue. O aprendiz utiliza a interface Criação / Carga Lista de Problemas para especificar de qual domínio, disciplina, ele quer que seja montada uma lista de problemas.



Figura 5.10 – Interface da Criação / Carga da Lista de Problemas

Os problemas existentes para um determinado perfil de usuário serão visualizado na interface Lista de Problemas.

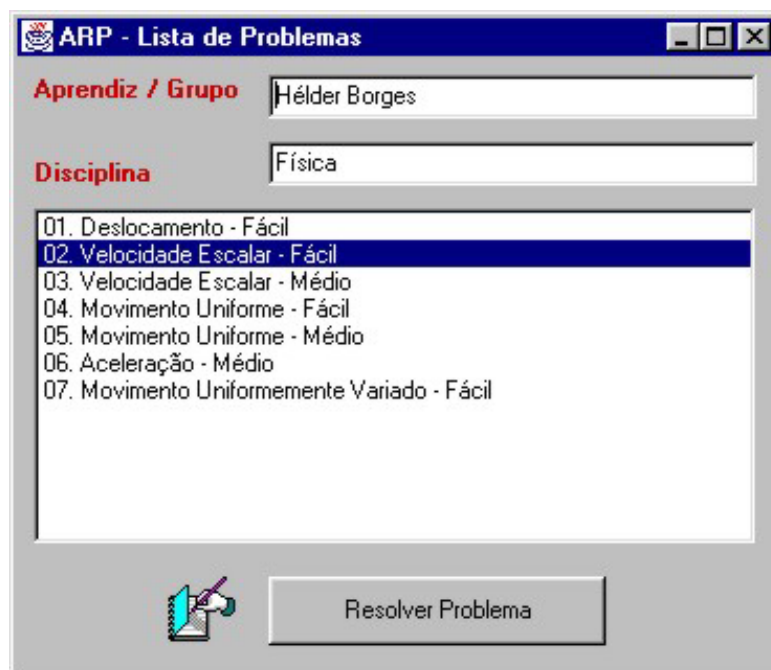


Figura 5.11 – Interface da Lista de Problemas

A partir da Interface Resolver Problemas, os passos designados para uma resolução de problemas são disparados, podendo-se especificar o objetivo, os dados, os conhecimentos

necessários e finalmente elaborar uma estratégia, como se pode ver nas quatro figuras seguintes.

Após o término de todas estas etapas, é gerada uma solução com as informações contidas em todas estas interfaces.

ARP - Resolver Problema

Aprendiz / Grupo Hélder Borges

Disciplina Física

Problema 02

Enunciado
Quantos kilometros um carro percorre estando com velocidade de 20 km/h se ele andou durante 3h e depois mais 7h com a mesma velocidade?

Especificar Objetivo [Icon] **Objetivo**

Especificar Dados [Icon] **Dados**

Especificar Conhecimentos [Icon] **Conhecimentos**

Elaborar Estratégia [Icon] **Estratégia**

Ajuda **Analisar Solução** **Resolva Problema**

Figura 5.12– Interface Resolver Problemas

ARP - Especificando Objetivo

Disciplina Física

Problema 02

Enunciado Quanto metros um carro percorre estando com velocidade de 20 km/h se ele andou durante 3h e depois mais 7h com a mesma velocidade?

Objetivo

Descrição Descobrir espaço

Variável S

Salvar

Figura 5.13 – Interface Especificar Objetivo

ARP - Especificando Dados

Disciplina Física

Problema 02

Enunciado Quanto metros um carro percorre estando com velocidade de 20 km/h se ele andou durante 3h e depois mais 7h com a mesma velocidade?

Dados

Descrição	Variável	Valor	Unidade
Velocidade	V	20	Km/h

✓ Adicionar

TEMPOA: A = 3 H
TEMPOB: B = 7 H

Salvar

Figura 5.14 – Interface Especificar Dados

ARP - Especificando conhecimentos

Disciplina Física

Problema 02

Enunciado Quantos kilometros um carro percorre estando com velocidade de 20 km/h se ele andou durante 3h e depois mais 7h com a mesma velocidade?

Conhecimentos

Conhecimento

V=S/T ✓ **Adicionar**

T=A+B

Salvar

Figura 5.15 – Interface Especificar Conhecimento

ARP - Elaborando estratégia

Disciplina Física

Problema 02

Enunciado Quantos kilometros um carro percorre estando com velocidade de 20 km/h se ele andou durante 3h e depois mais 7h com a mesma velocidade?

Estratégia

Passos

RESULTADO: S=200 ✓ **Adicionar**

Salvar

EQPRINCIPAL: V=S/T
MANIPULACAO: S=v*T
ENCONTRAR: T
EQUACAO: T=A+B
VALORES: T=3+7
RESULTADO: T=10
VALORES: S=20*10

Esboço

Drawing tools: Arrow, Text (A), Line, Square, Circle

Figura 5.16 – Interface elaborar Estratégia

5.1.5 Detalhamento dos Requisitos Não-Funcionais

Os requisitos não-funcionais incluem os requisitos de desempenho e atributos de qualidade, sendo em sua maioria globais (em relação ao Assistente e não ao Sistema Mathnet como um todo).

Alguns requisitos não-funcionais do Assistente de Resolução de Problemas são demonstrados nas tabelas que seguem

As operações de Cadastro de Problemas, Resolução de Problemas e Análise de Solução não poderão gastar mais tempo do que o valor máximo especificado pelo Sistema.

Tabela 5.7 – Requisito de Desempenho

O Assistente e Resolução de Problemas deve ser desenhado de forma que seja possível a sua expansão para um Agente Móvel

Tabela 5.8 – Requisito de Desenho

O Assistente deve restringir o acesso aos problemas conforme o nível do aprendiz.

Tabela 5.9 – Atributo de Qualidade (Segurança)

5.1.6 Classificação dos Requisitos

Todos os requisitos cadastrados recebem um identificador único dentro do projeto, o que permite sua referência nos documentos de descrição.

Nº	Nome Requisito	Tipo	Importância	Complexidade	Estabilidade
1	Interface do Professor	Interface	Essencial	Média	Média
2	Interface do Aprendiz	Interface	Essencial	Média	Alta
3	Caso de Uso Cadastro Problema	Caso de Uso	Essencial	Média	Alta
4	Caso de Uso Lista Problema	Caso de Uso	Essencial	Média	Alta
5	Caso de Uso Resolve Problema	Caso de Uso	Essencial	Alta	Alta
6	Caso de Uso Análise de Solução	Caso de Uso	Essencial	Alta	Alta
7	Requisito de Desempenho	Não- Funcional	Desejável	Média	Média
8	Requisito de Desenho	Não- Funcional	Essencial	Média	Média

Tabela 5.11 – Alguns Requisitos Classificados

5.1.7 Revisão dos Requisitos

O objetivo da revisão dos requisitos assegurar que a Especificação de Requisitos do Software tenha satisfeito os seguintes aspectos:

- Conformidade nos padrões e nos outros padrões já existentes;
- Atenda aos critérios de qualidade dos requisitos;
- Forneça informações suficientes para o desenho do Assistente, dos seus testes.

5.2 Estrutura do Assistente Resolução de Problemas

O Assistente de resolução de problemas possui duas interfaces principais. Uma para o professor cadastrar os problemas referentes à sua disciplina; outra, para o aluno, onde ele solicitará sua lista de problemas, escolherá um problema para resolver, executará os procedimentos de resolução, podendo solicitar ajuda, caso julgue necessário e, finalmente, submeterá a solução encontrada para verificação.

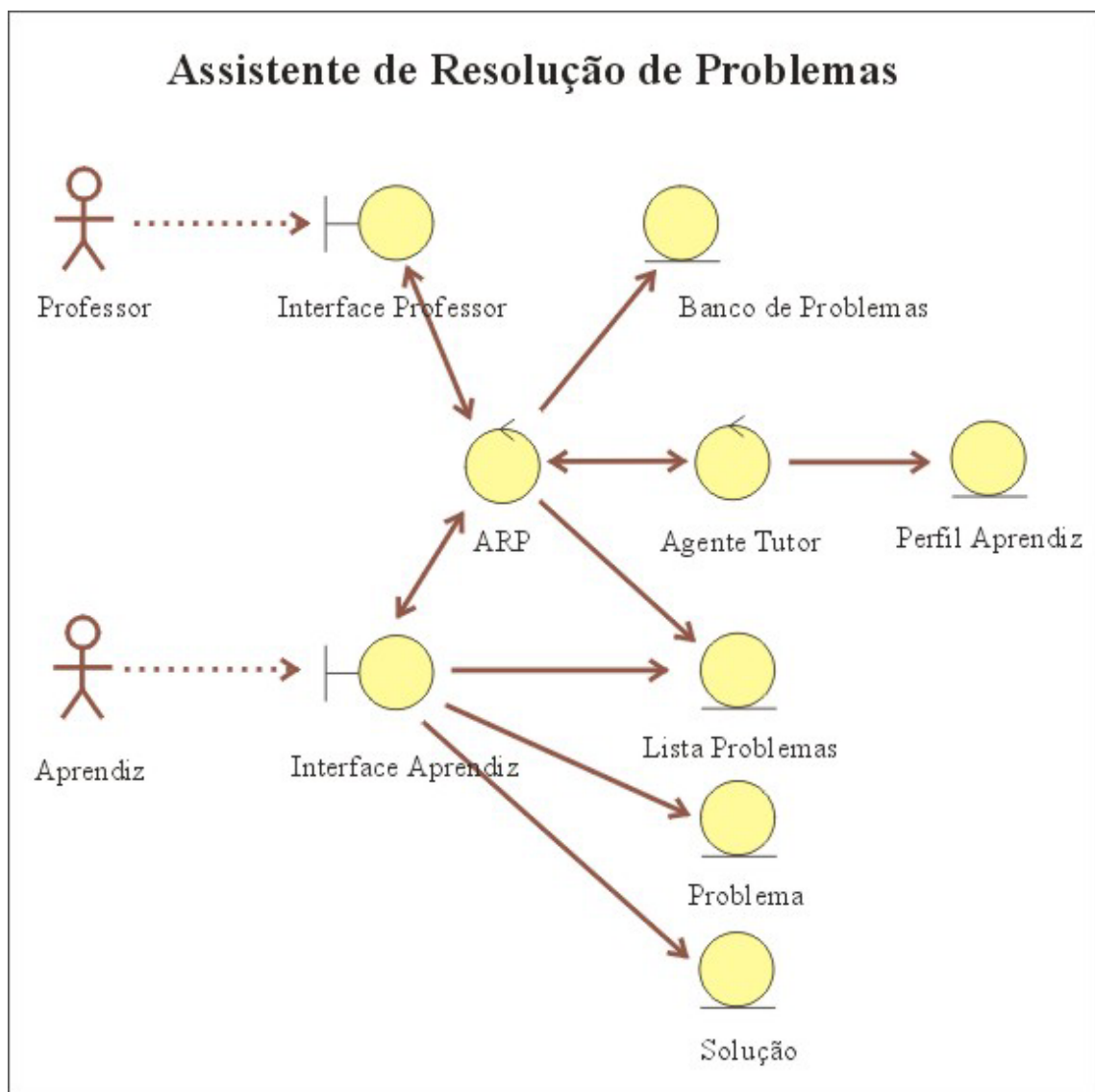


Figura 5.17 – Visão Geral das Classes que compõe o ARP

A estrutura do Assistente de Resolução de Problemas encontra-se apresentada na Figura 5.17.

O núcleo do Assistente é conhecido como ARP, contém as funções que resolvem e analisam problemas e é responsável por montar a lista de problemas para os aprendizes, em conformidade com o perfil do aprendiz informado pelo Agente Tutor que, por sua vez, foi informado pelo Agente Modelagem do Aprendiz.

5.2.1 Classe Problema

Demonstrada na Figura 5.18, é uma abstração dos dados referentes a um problema, os quais servem ao método Solucionar como base na formulação do processo de resolver problema.

É utilizada pelo ARP quando solicitado a resolver um determinado problema.

Problema
• Enunciado • Objetivo • Dados Iniciais • Conhecimentos • Nível
+ Solucionar

Figura 5.18 – Classe Problema

O agente humano, professor, responsável por um dado domínio, é incumbido da tarefa de cadastrar problemas referentes aos assuntos que ministrou.

Através de uma interface os problemas são cadastrados juntamente com as especificações requeridas pelo sistema para sua resolução, e são armazenados em uma base apropriada utilizando-se da classe problema.

5.2.2 Classe Solução

Também se constitui em uma abstração de dados, só que referentes a uma solução. É gerado enquanto o aprendiz resolve um problema e é utilizada pelo ARP quando vai analisar uma determinada solução. A classe Solução é Demonstrada na Figura 5.19.

Solução
• Problema • Dados Iniciais • Conhecimentos • Etapas • Resultado
+ Verificar

Figura 5.19 – Classe Solução

5.2.3 Diagrama de Seqüência

Cada caso de uso pode ter várias formas corretas de acontecerem, o que corresponderá a diferentes tipos de roteiros que são, em sua essência, seqüências de ações que exprimem o comportamento adotado.

Os diagramas de seqüência são orientados para exprimir o ordenamento temporal das ações de um comportamento.

O diagrama de seqüência do Assistente na Resolução de Problemas pode ser observado na Figura 5.20.

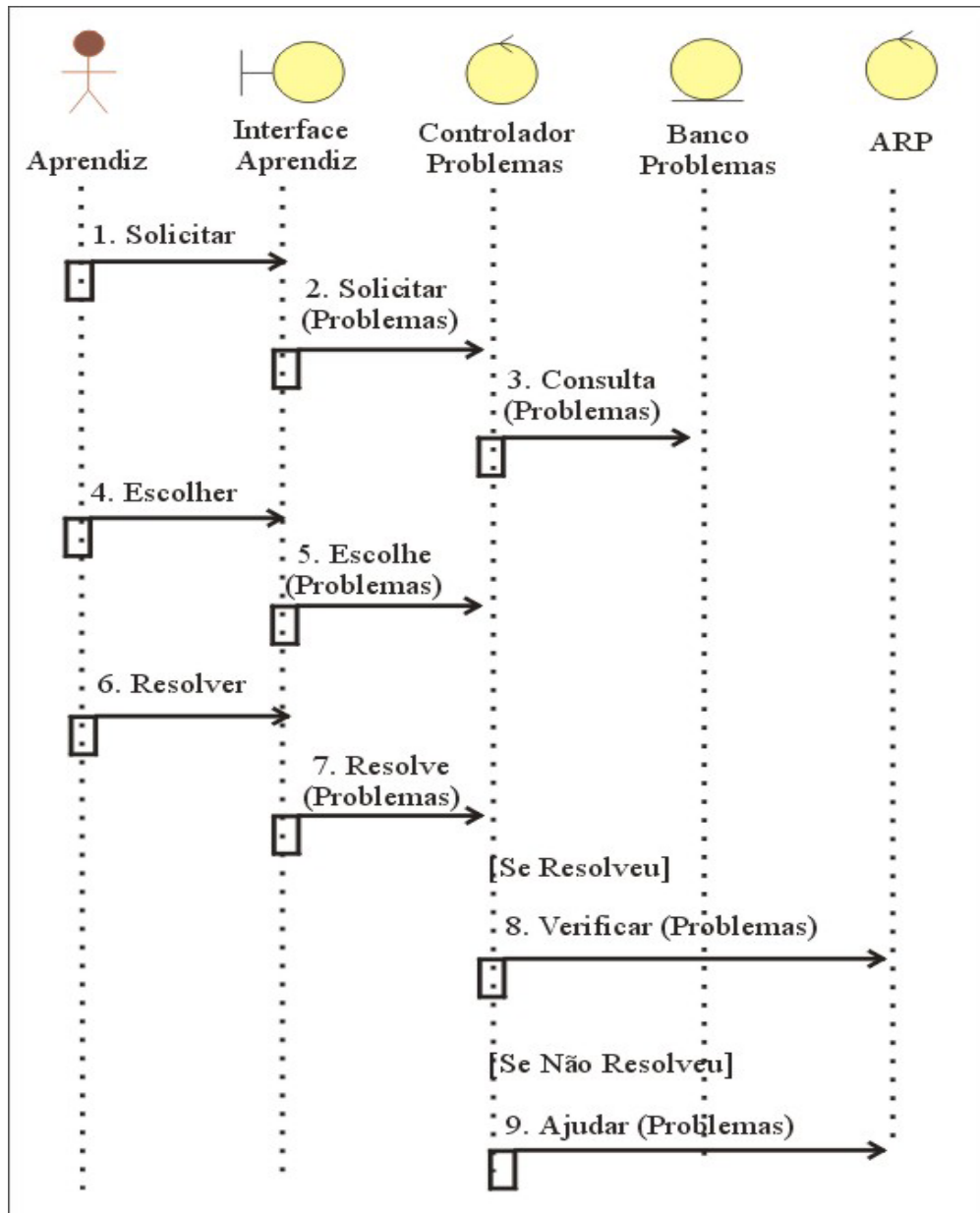


Figura 5.20 – Diagrama de Seqüência da Resolução de Problemas

5.3 Processo de Resolução de Problemas

A filosofia adotada pelo MATHNET para resolução de problemas não utiliza em sua totalidade nenhum dos métodos de resolução de problemas apresentado anteriormente. Ocorrendo o uso de etapas do modelo do George Pólya, bem como etapas do modelo do Serway, também poderemos utilizar, na formulação gráfica do resultado final, uma representação fazendo uso da técnica de mapas mentais. A opção por não utilizar um único modelo se faz necessário devido à adaptação à nossa realidade, levando em consideração tanto a complexidade na implementação, quanto do escopo em que estamos inseridos.

5.3.1 Aluno Resolvendo Problemas

O processo de resolver problemas por parte do aluno pode ser observado através do diagrama de colaboração, onde está detalhado cada etapa envolvida e as respectivas iterações entre as classes na Figura 5.21.

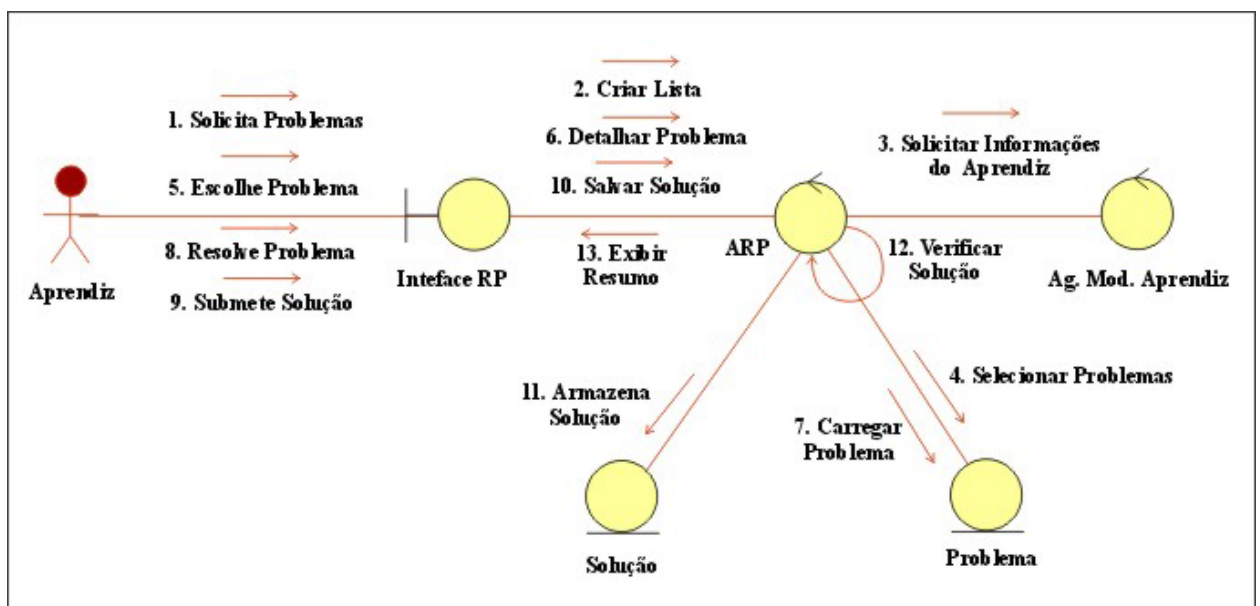


Figura 5.21 – Aluno Resolvendo Problema

Na Figura 5.22, temos o detalhamento da etapa 8, mostrada na figura anterior e que diz respeito à tarefa específica de resolver problemas. Podemos observar que existem algumas interfaces que, em conjunto, formam o que designamos de Interface de Resolução de problemas do Aprendiz.

Cada sub-interface tem funções específicas e são disparadas pela interface principal de Resolução de Problemas.

O aprendiz poderá ainda, quando não estiver conseguindo resolver o problema, solicitar ao sistema que o auxilie, indicando uma possibilidade a seguir. Quando solicitado para este propósito, o assistente verifica todas as etapas já especificadas pelo aluno para, então, sugerir um próximo passo.

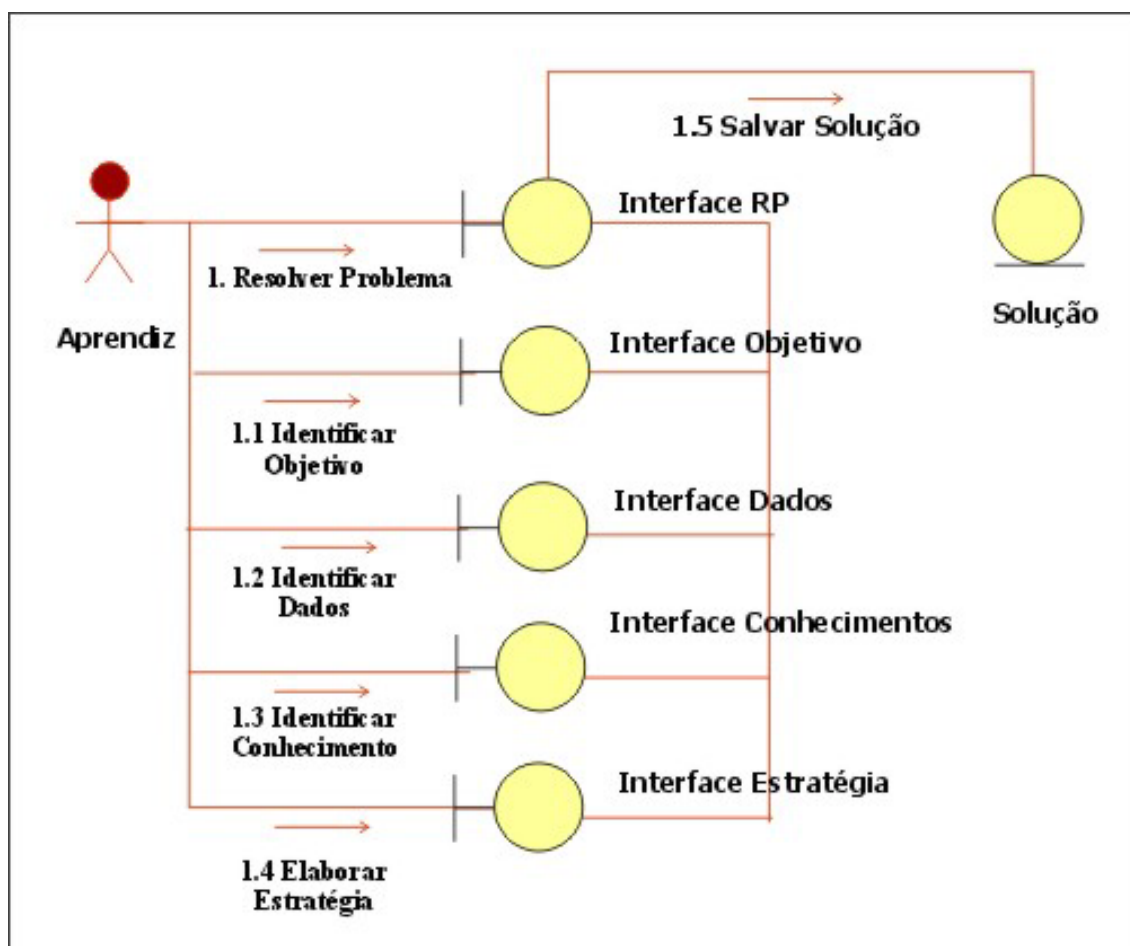


Figura 5.22 – Aluno Resolvendo Problema

5.3.2 Assistente Resolvendo Problemas

O Assistente pode resolver problemas em duas situações. A primeira, quando solicitado pelo aprendiz, onde fará uma solução completa; a outra situação acontece quando o aluno solicita ajuda devido não saber como prosseguir. Neste caso, o Assistente cria uma solução para o problema e analisa os passos que foram desenvolvidos pelo estudante, verificando se estão corretos e sugere um próximo passo.

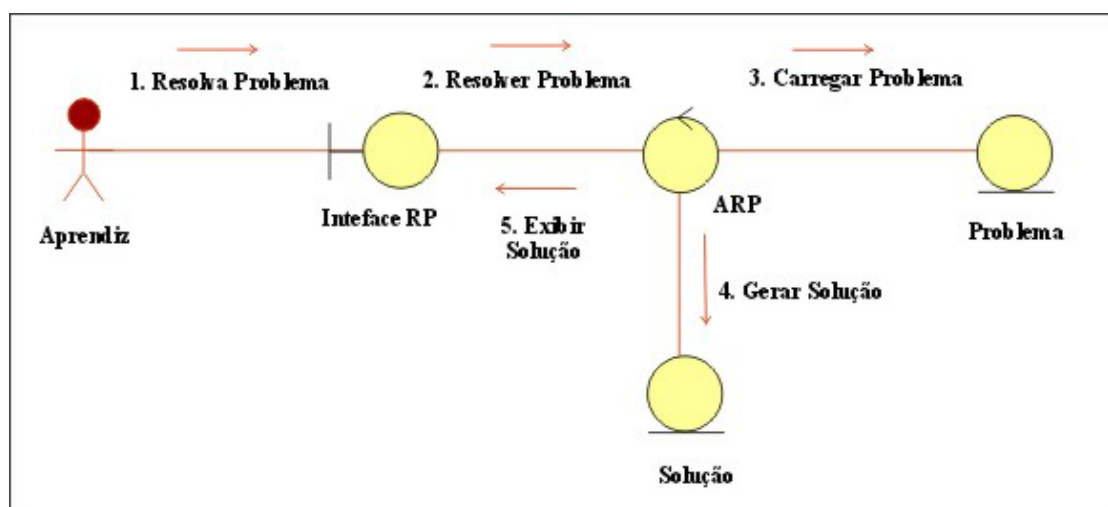


Figura 5.23 – Assistente Resolvendo Problema

5.3.3 Assistente Analisando Solução

Após o aprendiz terminar a última etapa na resolução de problemas, que é a elaboração da estratégia, a sua solução é salva e submetida a uma análise pelo Assistente de Resolução de Problemas, visando a verificar a consistência das afirmações informadas ali.

O processo de análise é composto por dois procedimentos. O primeiro envolve a verificação do que foi identificado como objetivo, tais como dados iniciais do problema e como conhecimentos necessários para resolução. Este primeiro processo é igual em todos os casos, pois, independente da estratégia adotada, estes valores permanecem sem alterações.

O segundo processo envolve a análise da estratégia desenvolvida, verificando-se a veracidade de cada etapa baseado nas informações corretas que foram obtidas pelo primeiro processo, ou seja, verifica-se se é possível afirmar cada etapa com os dados iniciais e conhecimentos especificados e se o resultado encontrado é compatível com o objetivo.

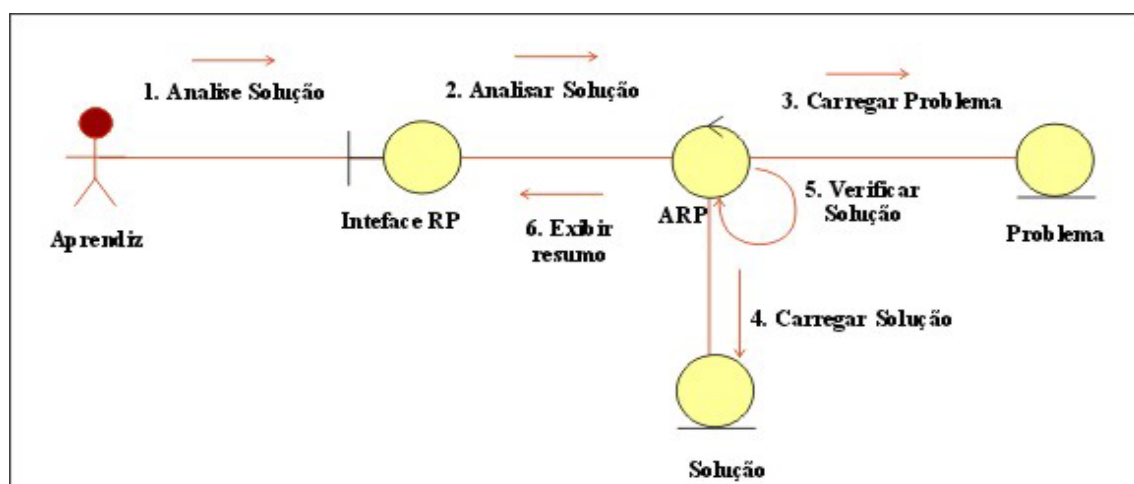


Figura 5.24 – Assistente Analisando uma Solução

5.4 Considerações Finais

O propósito deste capítulo foi descrever o comportamento do Assistente de Resolução de Problemas do Mathnet, bem como sua estrutura, suas restrições, funcionalidades. Sendo sua implementação realizada através da linguagem de programação Java. As razões que levaram a implementação do Assistente de Resolução de Problemas, em Java (Silva, 1997), foram duas: a primeira foi a interação com outros módulos do projeto Mathnet, que estão implementados em Java (Jeon, 2000); o segundo foi por ser totalmente orientada a objetos. As interfaces principais do ARP também foram demonstradas.

No próximo capítulo, apresentaremos os resultados obtidos com o ARP. Infelizmente, não poderemos estabelecer um comparativo com outras ferramentas análogas devido à indisponibilidade de alguma. Portanto, descreveremos simplesmente os resultados obtidos em nossos testes.

6. RESULTADOS OBTIDOS

Os resultados obtidos com a implementação do Assistente Resolução de Problemas foram animadores devido à alta complexidade apresentada nos processos de análise, modelagem e desenvolvimento, sendo isto constatado através da obtenção de corretas resoluções de problemas e análises de soluções.

Os processos de resolução de problemas e análise de soluções, como foi especificado anteriormente, tem seu escopo reduzido ao domínio da Física e ainda possuem restrições quanto à disposição dos elementos dos problemas. Isto acontece principalmente devido a grande quantidade de manipulações possíveis.

Foram executados diversos testes com a ferramenta, mas deve-se ainda submetê-la a outras baterias de testes visando a checar e comprovar o bom funcionamento de suas funcionalidades.

O fato de não termos disponível e até mesmo desconhecermos uma ferramenta que desempenhe o mesmo papel da nossa ferramenta, ou pelo menos semelhante, impossibilitou uma apuração mais precisa e até mesmo crítica dos resultados obtidos com o Assistente de Resolução de Problemas do Mathnet.

As figuras mostradas neste capítulo foram obtidas a partir dos resultados produzidos com a execução isolada do Assistente de Resolução de Problemas no aplicativo JGrasp, (JGrasp, 2002), que roda em cima do compilador Java, ou seja, para não rodar simplesmente via Prompt do DOS, usou-se este programa que possui uma interface mais amigável e diversas funcionalidades que auxiliam durante o processo de desenvolvimento e testes.

As figuras realçam cada as etapas principais, tanto do processo de resolução de problemas quanto de análise de solução.

6.1 Resolução de Problemas

Alguns problemas foram criados para testar o Assistente de Resolução de Problemas. Sendo que estes problemas possuem diferentes níveis de dificuldade. Dentre os problemas testados, todos tiveram o seu resultado corretamente obtido.

Os problemas para o ARP resolver estão armazenados em arquivos XML(Extensible Markup Language), que é uma linguagem utilizada para estruturar dados. Ou seja, os arquivos que armazenam os problemas estão formados via tags, como é definido em qualquer arquivo XML.

O desempenho do ARP quando resolve os problemas, em relação ao tempo, foi considerado satisfatório, solucionando rapidamente os problemas e satisfazendo os requisitos não-funcionais especificados na seção 5.1.5.

Nas figuras que seguem, podem ser visto os resultados do Assistente de Resolução de Problemas.

----jGRASP exec: Java Problema P1.xml
Problema: Qual a velocidade de um carro que percorreu 400 km em 5h?
Objetivo: Descobrir velocidade = V
Dados Iniciais:
Espaço → S=400km; Tempo → T = 5h;
Conhecimentos:
T=S/V;
EQPRINCIPAL: T=S/V
MANIPULACAO: V=S/T
VALORES: V=400/5
RESULTADO: V=80.0
----jGRASP: operation complete.

Figura 6.1 – Exemplo Resolução Problema 1

As figuras foram editadas, somente em nível de ilustração, para realçar os dois processos principais da resolução de problemas, ratificando-se que o conteúdo apresentado pelo ARP são os mesmos contidos nas figuras, porém sem os realces.

Temos destacado o enunciado do problema em branco, a especificação do objetivo, dos dados iniciais e dos conhecimentos com o cinza escuro, e a estratégia adotada em cinza mais claro.

----jGRASP exec: Java Problema P2.xml

Problema: Quantos kilometros um carro percorre estando
Com velocidade de 20 km/h se ele andou durante
3h e depois mais 7h com a mesma velocidade?

Objetivo: Descobrir espaço = S

Dados Iniciais:
Velocidade → V=20km/h; TempoA → A = 5h;
TempoB →B=7h;

Conhecimentos:
V=S/T; T=A+B;

EQPRINCIPAL:V=S/T
MANIPULACAO:S=V*T
ENCONTRAR:T
EQUACAO:T=A+B
VALORES:T=3+7
T=10;
VALORES:S=20*10;
RESULTADOS:S=200.0

----jGRASP: operation complete.

Figura 6.2 – Resolução Problema 2

----jGRASP exec: Java Problema P4.xml

Problema: Um avião voa 3h com velocidade de 900 km/h e nas 2 horas seguintes vai com 1000 km/h e retorna a velocidade inicial por mais 1h, qual a distancia percorrida por este aviao?

Objetivo: Descobrir $\text{espaco} = S$

Dados Iniciais:

Velocidade1 $\rightarrow B=900\text{km/h}$; Velocidade2 $\rightarrow C=1000\text{km/h}$;

Tempo1 $\rightarrow D = 3\text{h}$; Tempo2 $\rightarrow E=2\text{h}$; Tempo3 $\rightarrow F=1\text{h}$;

Conhecimentos:

$G=B*I$; $H=C*E$; $I=F+D$; $S=G+H$;

EQPRINCIPAL: $S=G+H$

ENCONTRAR: G

EQUACAO: $G=B*I$

ENCONTRAR: I

EQUACAO: $I=F+D$

VALORES: $I=1+3$

$I=4$;

VALORES: $G=900*4$;

$G=3600$;

ENCONTRAR: H

EQUACAO: $H=C*E$

VALORES: $H=1000*2$

$H=2000$;

VALORES: $S=3600+2000$

RESULTADO: $S=5600.0$

----jGRASP: operation complete.

Figura 6.3 – Exemplo Resolução Problema 3

6.2 Verificação de Soluções

As soluções testadas pelo Assistente de Resolução de Problemas também obtiveram um bom êxito nos resultados alcançados, exprimindo corretamente a análise sobre as soluções verificadas.

O resultado da verificação é produzido em relação a cada etapa da solução, sendo observada a veracidade das informações emitidas ali.

Se for considerado pelo ARP, como verdadeiro, ou seja, possível de deduzir, a etapa em questão é assinalada como OK; se não, uma mensagem indicando o erro é adicionada, passando-se então para a próxima etapa e, assim, consecutivamente, até o final.

Cada etapa é considerada individualmente, sendo analisado pelo Assistente se as implicações existentes são cabíveis, levando-se em consideração os dados iniciais e conhecimentos relativos ao problema.

```
----jGRASP exec: Java Solucao S1.xml  
  
OS DADOS ESTÃO CORRETOS.  
OS CONHECIMENTOS ESTAO CORRETOS.  
O OBJETIVO ESTA CORRETO.  
  
EQPRINCIPAL:T=S/V   OK!!!  
MANIPULACAO:V=S/T   OK!!!  
VALORES:V=400/5     OK!!!  
RESULTADO:V=80.0     OK!!!  
  
----jGRASP: operation complete.
```

Figura 6.4 – Solução Correta do Problema 1

Nas mesmas condições operacionais em que foram obtidas as figuras demonstradas no item 6.1, e sendo também relacionadas respectivamente aos mesmos problemas, poderemos observar nas figuras que seguem, o resultado obtido pelo Assistente de Resolução de Problemas. Na verificação de soluções, apenas foram retirados os comentários.

```
----jGRASP exec: Java Solucao S1.xml

OS DADOS ESTÃO CORRETOS.
OS CONHECIMENTOS ESTAO CORRETOS.
O OBJETIVO ESTA CORRETO.

EQPRINCIPAL:V=S/T  OK!!!
MANIPULACAO:S=V*T  OK!!!
EQUACAO:T=A+B  OK!!!
VALORES:T=3+7  OK!!!
T=10;  OK!!!
VALORES:S=20*10;  OK!!!
RESULTADOS:S=200.0  OK!!!

----jGRASP: operation complete.
```

Figura 6.5 – Solução Correta do Problema 2

```
----jGRASP exec: Java Solucao S1.xml

OS DADOS ESTÃO CORRETOS.
OS CONHECIMENTOS ESTAO CORRETOS.
O OBJETIVO ESTA CORRETO.

EQPRINCIPAL:T=S/V  OK!!!
MANIPULACAO:V=T/S  INCORRETO!!!
VALORES:V=40/5  INCORRETO!!!
RESULTADO:V=60.0  INCORRETO!!!

----jGRASP: operation complete.
```

Figura 6.6 – Solução Errada do Problema 1

Na figura anterior e na seguinte, são destacados erros detectados pelo ARP que foram praticados pelo Aprendiz na resolução dos problemas que foram descritos anteriormente.

```
----jGRASP exec: Java Solucao S1.xml

OS DADOS ESTÃO CORRETOS.
OS CONHECIMENTOS ESTAO CORRETOS.
O OBJETIVO ESTA CORRETO.

EQPRINCIPAL:V=S/T OK!!!
MANIPULACAO:S=V*T OK!!!
EQUACAO:T=A+B OK!!!
VALORES:T=3+7 OK!!!
T=10; OK!!!
VALORES:S=20*10; OK!!!
RESULTADOS:S=100.0 INCORRETO!!!

----jGRASP: operation complete.
```

Figura 6.6 – Solução Errada do Problema 2

6.3 Considerações Finais

O que obtivemos como resultado, foi profundamente positivo, considerando-se todas as dificuldades encontradas desde o início do processo, no que diz respeito, principalmente, a bibliografias específicas sobre resolução de problemas formais computacionalmente.

Pelo fato de termos desenvolvido uma ferramenta que resolve problemas e analisa soluções com relativa eficiência, entendemos lograr êxito em nossa proposta. Portanto, no próximo capítulo descreveremos nossas conclusões sobre resoluções de problemas no Mathnet e propostas para trabalhos futuros.

7. CONCLUSÃO

Considerando-se o caráter pedagógico de qualquer mecanismo de ensino, se faz necessária a existência de uma maneira de avaliar o grau de aprendizagem dos alunos em relação ao que está sendo ensinado. Mas é preciso lembrar que avaliação nada tem a ver com punição, considerando-se que o processo de avaliar serve, para identificar possíveis falhas na absorção do conteúdo pelo aluno, e com isto ajudar especificamente nestes pontos e também para identificar equívocos nos métodos pedagógicos empregados, para também corrigi-los e, então, melhorar o ensino.

A prática de resolver problemas é bastante utilizada nas escolas e isto implica que os alunos, de modo geral, já estão acostumados ou até mesmo possuem alguma afinidade com esta atividade. Com frequência, as escolas atribuem um caráter majoritário as atividades de resolução de problemas nas avaliações.

O Sistema Mathnet necessitava de uma ferramenta que resolvesse problemas e analisasse soluções para auxiliar o aprendiz no processo de fixação dos conteúdos ministrados e ainda contribuísse com seus resultados, especificamente em atividades de avaliação, na construção de uma avaliação geral, que envolve diversos critérios e que é iniciada na atividade pedagógica de apresentação do conhecimento.

Nesta dissertação, descrevemos o processo de análise e modelagem de um Assistente de Resolução de Problemas para o ambiente MATHNET de ensino aprendizagem cooperativa computadorizada.

Muitas dificuldades foram encontradas, principalmente no que se diz respeito ao levantamento bibliográfico. Existem muitos artigos e informações sobre resolução de problemas (matemáticos, físicos, etc.), porém, relacionadas com resolução de problemas acadêmicos em Sistemas de Ensino, são precárias as informações.

Avaliamos métodos de resolução de problemas para podermos definir uma metodologia de resolver problemas que fosse amigável ao aprendiz e também estruturável, considerando-se a necessidade lógica de manipulação computacional.

Levantamos os requisitos de um Assistente de Resolução de Problemas, especificamos suas funções e definimos sua estrutura. Também, definimos uma modelagem tanto para os problemas quanto para as soluções.

Na construção do Assistente de Resolução de Problemas, limitações relacionadas com a complexidade das tarefas de resolver problemas e analisar soluções. Devido às muitas possibilidades de caminhos a seguir, se fez necessário reduzir o escopo de abrangência do Assistente implementado, estabelecendo sua aplicabilidade unicamente a problemas de física e, ainda assim, impondo restrições.

Utilizou-se a linguagem Java no desenvolvimento do Assistente, em razão da compatibilidade com os demais módulos já existentes no Sistema MATHNET, que foram desenvolvidos nela e pelo fato de ser orientada a objeto.

Podemos destacar as seguintes vantagens da utilização do Assistente de Resolução de Problemas do Sistema Mathnet:

- Auxiliar o processo de avaliação geral, compondo um dos critérios estabelecidos pelo Sistema Mathnet;

- Auxiliar o aprendiz no processo de fixação de conteúdo, através da prática de resolver problemas;
- Emitir ajuda para o aprendiz, quando este solicita, enquanto ele resolve problemas;
- Resolução de problemas e Análise de Soluções automáticas, sem interferências externas.

No avanço da pesquisa, o Assistente de Resolução de Problemas poderá buscar as seguintes melhorias:

1. Em se tratando do escopo do Assistente, uma generalização deste, visando a abranger outros domínios diferentes da Física e que, por consequência, possuem diferentes tipos de problemas e soluções, dando com isto maior respaldo à ferramenta.
2. Aperfeiçoar o Assistente de Resolução de Problemas transformando-o em um agente inteligente móvel, ganhando-se com isto todos os benefícios inerentes a um agente deste tipo.
3. Criação de uma funcionalidade baseada em mapas mentais, que seja capaz de gerar um mapa mental a partir de uma solução, criando-se uma forma de melhor recordar e entender como se chegou àquela solução.

Com esta dissertação, esperamos ter contribuído para o estudo de Resolução de Problemas, entretanto, em nenhum momento pretendeu-se ter a palavra final sobre este assunto.

BIBLIOGRAFIA

- (Booch, 1999) BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **The Unified Modeling Language User Guide**. Addison-Wesley, 1999.
- (Bull et al., 1997) BULL, S., SMITH, M. **A Pair of Student Models to Encourage Collaboration**. In Proceedings of the Sixth International Conference (UM97). Vienna, New York: Springer Wien New York, 1997.
- (Buzan, 1974) BUZAN, Tony. **Use your head**. Editora BBC Books. 1ª Edição – 1974
- (Breuker et al., 1994) BREUKER, Joost., VELDE, Walter. **CommonKADS Library for Expertise Modelling**. IOS Press, Wasington, 1994.
- CHAIBEN, H. **Inteligência Artificial na Educação**. <http://www.cce.ufpr.br/~hamilton/iaed/iaed.htm>. Julho 2001.
- (Costa, 1997a) COSTA, E. de B. **Um Modelo de Ambiente Interativo de Aprendizagem Baseado numa Arquitetura Multi-Agentes**. Campina Grande, Tese de Doutorado – Universidade Federal da Paraíba.
- (Costa,1997b) COSTA, Evandro de Barros. **Um modelo de Ambiente Interativo de Aprendizagem Baseado numa Arquitetura Multi – Agentes**. Campina Grande, Paraíba, Brasil,1997.
- (Costa, 1999) COSTA, E. de B., SILVA, Josenildo C. and Ferneda, E. ***Designing and Development of an Authoring Environment for Building and Maintaining a Society of Artificial Tutoring Agents***. In Proceeding of SBIE'99, Curitiba, novembro de 1999.
- (Coutinho, 1999) COUTINHO, L. R. **A modelagem do aprendiz em ambientes de aprendizagem por computador baseados em atividades de resolução de problemas**. Universidade Federal da Paraíba, CURSO DE PÓS-GRADUAÇÃO EM INFORMÁTICA, Campina Grande, PB, 07 de maio de 1999.

- (Coutinho et al., 2000) COUTINHO, L., LABIDI, S., SERRA Jr, G., TEIXEIRA, C. ***A Learner Modeling Agent for Cooperative Learning***. Anais do XI Simpósio Brasileiro de Informática na Educação (SBIE'2000). Maceió-AL: 18 a 20 de Novembro de 2000.
- (Castelfranchi, 1995) CASTELFRANCHI, C. **Guarantees for autonomy in cognitive agent architecture**. In: Wooldridge, M. and Jennings, N. R., editors, Intelligent Agents: Theories, Architectures and Languages (LNAI Volume 890). Springer- Verlag: Heidelberg, Germany. 1995.
- COXFORD, F., SHULTE, P. **As idéias da álgebra**. São Paulo, Atual, 1995.
- (Davis, 1995) DAVIS, P., HERSH, R. **A Experiência Matemática**. Gradiva Publicações, 1995.
- (Elsom-Cook, 1993) ELSOM-COOK, M. 1993. **Student modeling in intelligent tutoring systems**. In Artificial Intelligence Review 7. Netherlands: Kluwer Academic Publishers.
- (Wilson, 2001) FILHO, Wilson. **Engenharia de Software: Fundamentos, Métodos e Padrões**. LTC Editora, 2001.
- (Ferber, 1988) FERBER, J.. M. **Problématiques des univers multi-agentes Intelligents. Actes des journées nationales**. Prc-greco Intelligence Artificielle, Toulouse, 1988.
- (Ferreira, 1998) FERREIRA, J. S. **Concepção de um Ambiente Multi-Agentes de Ensino Inteligente Integrando o Paradigma de Aprendizagem Cooperativa**. Dissertação de Mestrado, Universidade Federal do Maranhão - UFMA. São Luís, 1998.
- (Finin, 1997) FININ T., LABROU Y.; MAYFIELD J. **KQML as an agent communication language**, Cambridge: MIT Press , 1997.
- (FIPA, 1997) FIPA. **Agent communication language**. Technical report. Foundation for Intelligent Physical Agents. 1997.
- (Furlan, 1988) FURLAN, José Davi. **Modelagem de Objetos através da UML** . São Paulo: Makron Books, 1998.
- FRIEDMAN, E. **JESS the Rule Engine for the Java Platform**. <http://herzberg.ca.sandia.gov/jess/main.html>. Junho 2001.

- (Gaines e Shaw, 1995) GAINES, Brian., SHAW, Mildred. **Collaboration through Concept Maps**. 1995. <http://ksi.cpsc.ucalgary.ca/articles/CSCL95CM/>. Fevereiro, 2002.
- (Gardner, 1994) GARDNER, Howard. **As estruturas da mente: a teoria das inteligências múltiplas**. Porto Alegre, Artes Médicas, 1994.
- (Gardner et al., 1989) GARDNER, H., HATCH, T. **Multiple intelligences go to school: Educational implications of the theory of multiple intelligences**. Educational Researcher, 1989.
- (Genesereth et al., 1999) GENESERETH, M. R.; KETCHPEL, S. P. **Software agents**. Communications of the ACM, v.37,nº 7. 1994.
- (Gilmore et al., 1988) GILMORE, D., SELF, J. **The application of machine learning to intelligent tutoring systems**. In J. Self (Ed.) Artificial Intelligence and Human Learning, Intelligent computer-assisted instruction, New York: Chapman and Hall, 1988.
- (Haydt et al., 1995) HAYDT, R., CAZAUX, C. **Curso de Didática Geral**. 2a edição, São Paulo: Editora Ática,1995.
- (Haydt, 1997) HAYDT, R. **Avaliação do Processo de Ensino-aprendizagem**. São Paulo: Editora Ática, 1997.
- HENRIQUES, P. 1997. **Ensino a distância suportado pela Internet**. <http://student.dei.uc.pt/~jpdias/sf/>. Agosto 2001.
- (Jeon, 2000) JEON, H. **JATLite: a java agent infrastructure with message routing**. IEEE Internet Computing, v. 4, nº 2. March-April 2000.
- (JGrasp, 2002) **Graphical Representations of Algorithms, Structures, and Processes**. Auburn University. <http://www.eng.auburn.edu/grasp/>. Janeiro 2002.
- (Júnior, 2000) JÚNIOR, Miguel. **XML**. Universidade Federal do Rio de Janeiro. http://www.gta.ufrj.br/grad/00_1/miguel/. Maio 2001.

- KOMATSU, R. 1997. **Aprendizagem Baseada em Problemas: Organizando Recursos Educacionais**. <http://www.famema.br/trabalhos/poster1/welcome.html>. Agosto 2001.
- (Kotz, 1999) KOTZ, David; GRAY, Robert S. **Mobile Code: The Future of the Internet**. In: WorkShop “Mobile Agents in the Context of Competition and Cooperative (MAC3)” at Autonomous Agents, May 1, 1999., Seattle, Washington, USA. <http://actcomm.dartmouth.edu/papers/kotz:future.ps.gz>. Setembro 2001.
- LANZING, J. **The Concept Mapping Homepage**. 1997. <http://users.edte.utwente.nl/lanzing/>. Junho 2001.
- (Labidi, 1998) LABIDI, S., FERREIRA, J. S. (1998). **Technology-Assisted Instruction Applied to Cooperative Learning: the SHIECC Project**. In Proceedings of the IEEE International Conference Frontiers in Education (FIE'98). Tempe, Arizona. November 4-7, 1998.
- (Labidi et al., 1998) LABIDI, S., FERREIRA, J. S. **Modelagem do Aprendiz no Quadro do Ensino Cooperativo**. Anais do Simposium Brasileiro em Informática na Educação (SBIE'1998). Fortaleza-CE. 1998.
- (Labidi et al., 2000a) LABIDI, S, JOSENILDO C. Silva, Luciano R. C., Evandro C. B. **MathNet: An Agent-Based Tutoring System for Supporting Cooperative and Distant Learning**. To appear in the Proceedings of the International Conference on Computers and Advanced Technology in Education (CATE'2000). May 24-27, 2000. Cancun, Mexico.
- (Labidi et al., 2000b) LABIDI, S., Sérgio N. **Student Modeling and Semi-Automatic Domain Ontology Construction for SHIECC**. In the Proceedings of the IEEE International Conference Frontiers in Education (FIE'2000). Kansas City, Missouri, USA. October 18-21, 2000.
- (Lange, 1998) LANGE, Danny B. **Mobile Objects and Mobile Agents: The Future of Distributed Computing? In Programming and Deploying Java™ Mobile Agents with Aglets™**, Lange and Oshima. Addison-Wesley, 1998.

- MCALLISTER, H. **Common Sense Problem Solving and Cognitive Research**.
<http://www2.hawaii.edu/suremath/cognitive.html>. Agosto 2001.
- (Moran, 2000) MORAN, José Manuel, MASETTO, Marcos e BEHRENS, Marilda. **Novas Tecnologias e Mediação Pedagógica**. São Paulo, Papirus Editora, 2000.
- (Nakamura et al., 1996) NAKAMURA, M., Takeuchi, A., OTSUKI, S. **Roles of Student Models for Assisting Group Learning**. In the Proceedings of the 5 th International Conference on User Modeling (UM'96). Kailuna-Kona, Hawaii, USA: 2-5 Jan. 1996.
- (Novak et al., 1996) NOVAK, Joseph., GOWIN Bob. **Aprender a Aprender**. Lisboa: Plátano. 1996.
- (Nunes, 2001) NUNES, Helena Maria Pereira. **Serviço de Busca baseado em Agentes Móveis para o Ambiente Mathnet de Ensino Cooperativo Computadorizado**. 2001, Xp Dissertação (em Ciência da Computação), UFMA, São Luís.
- (Nunes et al., 2000) NUNES, H., LABIDI, S. **Increasing Interactions within MATHNET using Mobile Agents**. To appear in 4th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2000) and The 6th International Conference on Information Systems Analysis and Synthesis (ISAS 2000), Orlando, Florida, USA, July 23-26, 2000.
- (Nwana et al., 1999) NWANA, H., NDUMU, D., LEE, L., COLLINS, J. **ZEUS: a toolkit forbuilding distributed multi-agent systems**. In Applied Artificial Intelligence Journal, Vol 13 (1), 1999.
- (Odell, 1999) ODELL, J. **Agents and objects: how do they differ?**, working paper v2.2. September, 1999.
- (Paraná, 1993) PARANÁ, D. **Física – Volume 1 – Mecânica**. São Paulo: Editora Ática, 1993
- (Pólya, 1945) PÓLYA, G., **How to Solve It**. Princeton, Princeton University Press, 1945.
- (Rao et al., 1991), RAO, A.S.; GEORGEFF, M.P. **Modeling rational agents within a BDI-architecture**. California: Morgan Kaufmann Publishers. 1991.

- (Rosa et al., 2001) ROSA, J., MONTEIRO, J., APARECIDA. S., CARDOZO. W. **Resolução de Problemas: Investigação, Ensino, Avaliação e Formação de Professores.** <http://educacaomatematica.vila.bol.com.br>. Junho 2001.
- (Rosechein, 1985) ROSENCHEIN, J.S., GENESERETH, M. R. **Deals among rational agents.** In International Joint Conference on Artificial Intelligence (IJCAI - 85), 9. California. 1985.
- (Russel et al., 1995) RUSSELL, S. J., NORVING P. **Artificial Intelligence: A Modern Approach.** Prentice Hall, 1995.
- SANTOS, N. **Estado da Arte em Espaços Virtuais de Ensino e Aprendizagem.** <http://www.inf.ufsc.br/sbc-ie/revista/nr4/070TU-santos.htm>. Junho 2001.
- SERRA JR., G. **Um Agente de Modelagem do Aprendiz para o Ambiente Mathnet de Ensino Inteligente Cooperativo Computadorizado.** Universidade Federal do Maranhão, Curso de Pós-Graduação em Engenharia Elétrica, São Luís, MA, 22 de setembro de 2001.
- (Serra Jr. et al., 2001) SERRA JR., G., LABIDI, S., NUNES, H. **Learner Modeling for Information Retrieval in the MATHNET Environment.** Proceedings of the 5 th World Multi-Conference on Systemics, Cybernetics and Informatics (SCI 2001). Orlando, Flórida – EUA: July 22-25, 2001.
- (Silva, 1997) SILVA, Luciano, GARCIA, Nilza Aréan. **Introdução à Linguagem Java.** Departamento de Ciência da Computação – IME/USP. Instituto de Pesquisas Tecnológicas – IPT. Fevereiro de 1997.
- SILVEIRA, J. **Resolução de problemas.** <http://athena.mat.ufrgs.br/~portosil/resu.html>. Julho 2001.
- (Sleeman et al., 1981) SLEEMAN, D., SMITH, M. **Modelling student's problem solving. Artificial Intelligence.** 1981. (Spensley, 1990) Spensley, F. *Using multiples teaching strategies in a ITS.* In C. Frasson & G. Gauthier (Eds) Intelligent Tutoring Systems: At the crossroads of Artificial Intelligence and Education. Norwood, NJ: Ablex, 1990.

- (Steinberg, 1984) STEINBERG, E. **Teaching Computers to Teach. Computer-based education Research.** Laboratory and College of Education University of Illinois at Urbana-Champaign.
- (Serway, 1996) SERWAY, R., **Física 4 para cientistas e engenheiros com física moderna,** Rio de Janeiro, LTC, 1996.
- (Slavin, 1990) SLAVIN, R. E. **Cooperative Learning: Theory, Research and Practice.** Prentice Hall, 1990.
- (Vygostky, 1978) VYGOSTKY, L. **Mind in Society: The Development of Higher Psychological Process.** Cambridge, MA: Harvard University Press, 1998.
- (Walnum, 1997) WALNUM, Cleyton. **Java em Exemplos.** São Paulo: Makron Books, 1997.
- (Wooldridge et al., 1995) WOOLDRIDGE, Michael., NICHOLAS R. Jennings. **Agent Theories, Architectures, and Languages: a Survey.** In Wooldridge and Jennings Eds., *Intelligent Agents*, Berlin: Springer-Verlag, 1995.
- ZANCANER, Carmen Lúcia Coube. **Como estudar melhor: a técnica do mapa mental.** Bauru, SP. Edições Tilibra, 1995.