

## DESENVOLVIMENTO DE UM *SOFTWARE* BASEADO NO CONCEITO DE OBJETOS DE APRENDIZAGEM APLICADO A MODELOS MATEMÁTICOS

**Rosely Maria Velloso Campos** – rcampos@pucminas.br

Pontifícia Universidade Católica de Minas Gerais, Instituto Politécnico  
30535-901 – Belo Horizonte - MG

**Daniel Augusto de Melo Santos** – danielmelos@gmail.com

Pontifícia Universidade Católica de Minas Gerais, Instituto de Informática  
30535-901 – Belo Horizonte - MG

***Resumo:** No presente trabalho utilizou-se o conceito de objetos de aprendizagem para desenvolver um software educacional direcionado para o ensino de modelos matemáticos. Tal software educacional contém as características funcionais de acessibilidade, reusabilidade e interoperabilidade, as quais beneficiam o professor, permitindo-lhe criar novas aplicações pedagógicas a partir dos componentes já desenvolvidos no software. Ele é, portanto, um típico objeto de aprendizagem, criado para estimular o aluno a ser criativo e descobridor de novas abordagens, as quais solucionem problemas matemáticos antes considerados complexos. O aluno terá o estímulo de comprovar, virtualmente, a relação teoria e prática, o que o estimulará a ser autodidata na construção da sua educação científica e tecnológica. O software educacional é composto de quatro componentes (Construtor de Equações Gráficas, Construtor de Diagramas em Bloco, Interpretador de Equações e Visualizador de Animações e um Sistema de Ajuda.*

***Palavras-chave:** Educação, Engenharia, Objeto de aprendizagem, Modelagem*

### 1 INTRODUÇÃO

O tema central deste trabalho é a pesquisa e desenvolvimento de um *software* educacional, que permita aos estudantes universitários de engenharia resolverem questões, consideradas por eles complexas, de modelagem matemática de sistemas físicos reais e/ou que os estimulem a explorar as potencialidades da tecnologia digital existente na área de informática.

Os notáveis avanços na área de informática têm permitido a exploração de novas aplicações em diversas áreas do conhecimento humano, como é do nosso conhecimento no dia a dia. É o caso da tecnologia de animação, que tem possibilitado a execução de filmes com efeitos especiais, os quais levam o mundo virtual a simular, de maneira convincente, ações humanas em ambientes da natureza real, criando uma realidade ficcional possível de realização. É o caso, também, da aplicação de recursos computacionais na área de medicina,

os quais permitem gerar animações de partes ou do todo de seções fatiadas do corpo humano, revelando detalhes anatômicos importantes nos estudos de diversos tipos de doenças. Um outro exemplo da aplicação de informática está na área de ensino, que é o foco deste trabalho.

Recentemente, tem aumentado o estudo e desenvolvimento de técnicas de ensino com recursos computacionais, cujos benefícios para a transmissão de conhecimentos na área acadêmica começam a ser explorados. É o caso da tecnologia envolvendo objetos de aprendizagem (*Learning Objects*) (WILEY,2005).

*Learning Objects* são entidades digitais projetadas para auxiliarem o aluno a atingir um determinado objetivo pedagógico. Tais objetos podem ser reutilizados em outros contextos de suporte à aprendizagem, o que se constitui numa vantagem econômica para os usuários desta tecnologia. É uma ferramenta de trabalho muito versátil, projetada, também, para possibilitar a transmissão de conhecimento em cursos de ensino à distância (EAD) ou de ensino semi-presencial (ESP). E aqui, novamente, tal tecnologia apresenta outra vantagem, caracterizada pela contribuição à diminuição dos níveis de analfabetismo no país bem como na elevação do nível de formação básica de estudantes de comunidades distantes e carentes (RIVED,2005).

O tema abordado neste trabalho é atual e de grande relevância por ser uma ferramenta voltada para a melhoria da aprendizagem. Como se sabe, investimento em políticas educacionais é visto como uma condição fundamental para o desenvolvimento sócio - econômico - cultural de uma nação. No presente trabalho, portanto, agrega-se elementos de inovação tecnológica, uma vez que envolve o desenvolvimento de um produto com maior valor agregado.

O *software* desenvolvido é, portanto, um típico objeto de aprendizagem. Como tal, ele é desenvolvido para ser lúdico e, ao mesmo tempo, ser uma ferramenta educacional reutilizável. Com estas características, o objeto de aprendizagem estimula o aluno a ser criativo e descobridor de novas abordagens, as quais solucionem problemas matemáticos antes considerados complexos. Assim, usando o objeto de aprendizagem, o aluno poderá:

- compor seus modelos matemáticos de forma fácil e intuitiva; e
- exibir os sistemas físicos com animações.

Deste modo, o aluno terá o estímulo de comprovar, virtualmente, a relação entre teoria e prática, o que o estimulará a ser autodidata na construção da sua educação científica e tecnológica.

O crescente interesse internacional no campo de objetos de aprendizagem reutilizáveis origina da percepção de que esta tecnologia inovadora tem um enorme potencial de criar novas técnicas de desenvolvimento e distribuição de conteúdo através da Internet. Não é de se estranhar, portanto, o crescente aumento em publicações procurando conceituar corretamente objetos de aprendizagem ou mesmo relatar novos objetos de aprendizagem, desenvolvidos com as seguintes características:

- *Software* contendo um jogo educativo, visando a aprendizagem de algoritmo genético (SILVEIRA & BARONE,1998);
- *Software* aplicado na visualização das transformações de uma câmera no ambiente 3D (LALEUF & SPALTER,2001);
- Elemento multimídia, também denominado “Aplicação Cliente”, desenvolvido com base no programa Flash, com enfoque no ensino de geometria plana (LIMA *et al*,2005);
- *Software* contendo um jogo educativo para o auxílio na alfabetização de crianças com hiperatividade (SILVA *et al*,2005); e
- Elemento multimídia, também denominado “Aplicação Cliente”, desenvolvido com base no programa Flash, visando a aprendizagem de geoprocessamento (MAEDA *et al*,2005).

No caso do Ensino Superior, a literatura mostra poucas iniciativas de aplicações de objetos de aprendizagem em áreas tais como a Física e a Matemática. Neste sentido, este

trabalho contribui na ampliação da aplicação dos objetos de aprendizagem, desenvolvendo um *software* em uma plataforma flexível, onde diversos modelos matemáticos poderão ser estudados. A utilização desta ferramenta de ensino pelo aluno facilitará:

- o entendimento dos conceitos, melhorando assim, substancialmente, a relação ensino-aprendizagem; e
- o trabalho cooperativo entre alunos, estimulando atividades de produção de conhecimento compartilhado.

Neste caso, o objeto de aprendizagem é, portanto, um *software* educacional composto com os seguintes recursos:

- um construtor de equações gráficas;
- um construtor de diagramas;
- um interpretador de equações;
- um visualizador de animações previamente elaboradas; e
- um sistema de ajuda.

A criação deste objeto de aprendizagem demandou uma interação multidisciplinar, uma vez que envolve aspectos de ciência da computação (linguagem de programação, algoritmos e estruturas de dados e análise de sistemas) e de engenharia eletrônica e de telecomunicações (modelagem matemática, circuitos e cálculo).

A seguir, detalhamos um estudo de caso, os resultados obtidos, as conclusões e sugestões para trabalhos futuros.

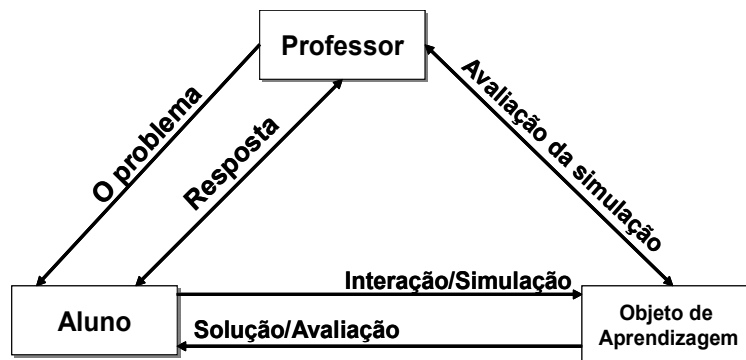
## 2 ESTUDO DE CASO

O número crescente de informações disponibilizadas nesta era da informática exige de nós, usuários, lançar mão de novas maneiras para assimilá-las e para consolidá-las de forma correta e rápida. Estas novas maneiras exigem tecnologias que mudem os patamares de interação com a realidade, permitindo uma comunicação através de sons, imagens e textos, num formato que integre mensagens e tecnologias multimídia. Este é, portanto, o cenário no qual a busca do desenvolvimento da tecnologia dos objetos de aprendizagem acontece.

No ensino superior, a cada dia os docentes percebem a necessidade de criar novos meios e ambientes de aprendizagem que venham a servir de motivação aos alunos. O ambiente virtual e semi-virtual surge como meio facilitador na aprendizagem do aluno, desenvolvendo uma atitude de parceria e co-responsabilidade entre professor-aluno desde o planejamento até a efetiva execução do plano de ensino, trabalhando com as dificuldades, a disciplina e autonomia de cada aluno. *Learning Objects* podem ser usados tanto no processo de aprendizagem, bem como no processo de avaliação, preocupando sempre com o processo de aprendizagem individual de cada aluno, com um caráter de diagnóstico (MASETTO, 2003).

Usar recursos da tecnologia de informática e dos objetos de aprendizagem atende a lógica de Vygotsky (VYGOTSKY,1989) que, segundo a qual, o pensamento é construído gradativamente num ambiente histórico e, em essência, social. A aplicação da abordagem de Vygotsky (VYGOTSKY,1989) na prática educacional requer que o professor reconheça a idéia de zona de desenvolvimento proximal e estimule o trabalho colaborativo, de forma a potencializar o desenvolvimento cognitivo dos alunos. As habilidades podem ser desenvolvidas com a ajuda de um professor que servirá como orientador ou pela colaboração de pares. Nesta abordagem, os professores são os principais agentes do processo educacional e devem ser capazes de dominar o uso de recursos com conhecimento e compreensão desse uso, de acordo com o assunto, assumindo, ainda, tarefas de comunicador e figura de identificação.

Portanto, os ambientes colaborativos de aprendizagem, apoiados em computadores e tecnologias associadas, no caso os novos objetos de aprendizagem, valorizam a abordagem de Vygostky (VYGOTSKY,1989), ao criar um espaço de trabalho conjunto entre professor e aluno. E isto pode ser feito a partir de uma situação autêntica, onde os alunos assumem a responsabilidade de planejar e desenvolver meta-habilidades cognitivas, monitorando e dirigindo a própria aprendizagem bem como avaliando o próprio desempenho. Atingir tais resultados é o que motiva o presente trabalho. Deste modo, a relevância deste trabalho é a de propor o desenvolvimento de um objeto de aprendizagem, visando facilitar a interação aluno – objeto de aprendizagem – professor segundo o fluxograma esquemático da Figura 1.



**Figura 1** – Logística de um objeto de aprendizagem.

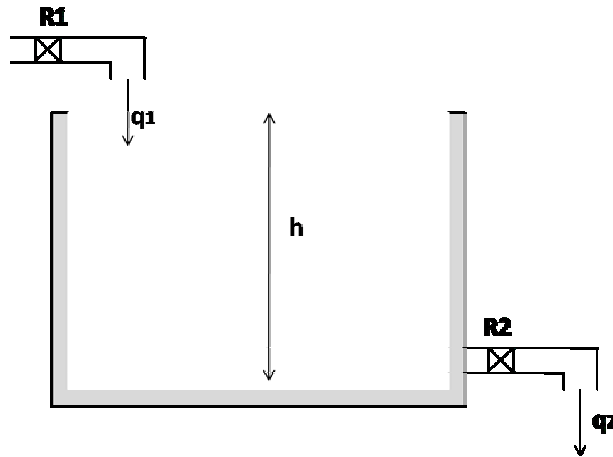
### 3 RESULTADOS OBTIDOS

No desenvolvimento desse *software* foi necessário seguir uma sequência de passos descritos a seguir:

- caracterizar o problema real (exemplos a serem simulados);
- planejar o *software* de modo a definir a linguagem de programação e a plataforma de desenvolvimento dos componentes do *software*;
- desenvolver os componentes do *software*
- gerar um sistema de ajuda;

#### 3.1 Caracterização do Problema Real

Nosso primeiro problema real a ser modelado é um sistema que consiste de um tanque de seção reta uniforme de área  $A$  ao qual é adaptada uma resistência ao fluxo,  $R$ , tal como uma válvula (torneira), uma tubulação ou vertedouro. Suponha que  $q_0(t)$ , a vazão volumétrica (volume/tempo) através da resistência, se relaciona com a altura de líquido,  $h(t)$ , pela relação linear  $q_0(t) = h(t)/R$ . Uma vazão volumétrica variável com o tempo,  $q(t)$ , de um líquido de massa específica constante,  $\rho$ , alimenta o tanque. Deseja-se determinar a função de transferência que relaciona o nível do líquido no tanque com a vazão de entrada. A Figura 2 representa, graficamente, esse primeiro problema real. Nela é mostrado um tanque de seção retangular, ao qual acoplamos duas torneiras, uma de entrada ( $R_1$ , considerada estar totalmente aberta) e outra de saída de água ( $R_2$ ), com suas respectivas vazões,  $q_1(t)$  e  $q_2(t)$ . Este tanque está com um nível variável de água,  $h(t)$ . Os valores de  $A$ ,  $q$  e  $R_2$  são dados.



**Figura 2** - Representação esquemática do primeiro problema real.

A função de transferência começa a ser determinada quando estabelece-se a relação entre o volume de água que entra e o que sai, de acordo com as equações 1 e 2:

$$A \frac{dh(t)}{dt} = q_1(t) - q_2(t) \quad (1)$$

$$q_2(t) = \frac{h(t)}{R_2} \quad (2)$$

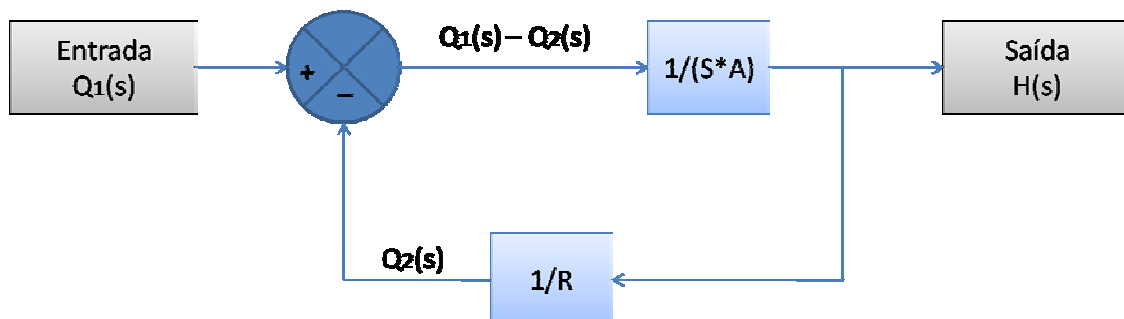
Combinando as duas equações, produzimos uma terceira, dada por:

$$A \frac{dh(t)}{dt} = q_1(t) - \frac{h(t)}{R_2} \quad (3)$$

Na equação 3, aplicamos a Transformada de Laplace, de modo a gerar uma nova equação (equação 4) no domínio da frequência:

$$AsH(s) = Q_1(s) - \frac{H(s)}{R_2} \quad (4)$$

Após aplicar a Transformada de Laplace, desenhou-se o diagrama em bloco desse problema representado pela Figura 3, facilitando a visualização da função de transferência desejada.



**Figura 3** - Diagrama em bloco do primeiro problema real.

Deste diagrama conclui-se que a função de transferência procurada é uma relação entre o nível da coluna  $H(s)$  e a vazão de entrada  $Q_1(s)$ . Logo, continua-se a resolução gerando novas equações (equações 5 a 7):

$$H(s)\left[As + \frac{1}{R_2}\right] = Q_1(s) \quad (5)$$

$$H(s)\left[\frac{AR_2s + 1}{R_2}\right] = Q_1(s) \quad (6)$$

$$\frac{H(s)}{Q_1(s)} = \frac{R_2}{(AR_2s + 1)} \quad (7)$$

Sendo a equação 7 a função de transferência (FT). A solução da função de transferência demanda fazer a seguinte pergunta: *Qual é o nível para uma vazão  $Q_1$  conhecida?* E a resposta é:  $q_1(t) = 1000$  l/h. Então, no domínio da frequência, o valor de  $Q_1$  pode ser definido por  $Q_1(s) = 1000/S$ . A resposta no domínio do tempo é dada pela equação 8:

$$h(t) = 100 - 100e^{-t/5} \quad (8)$$

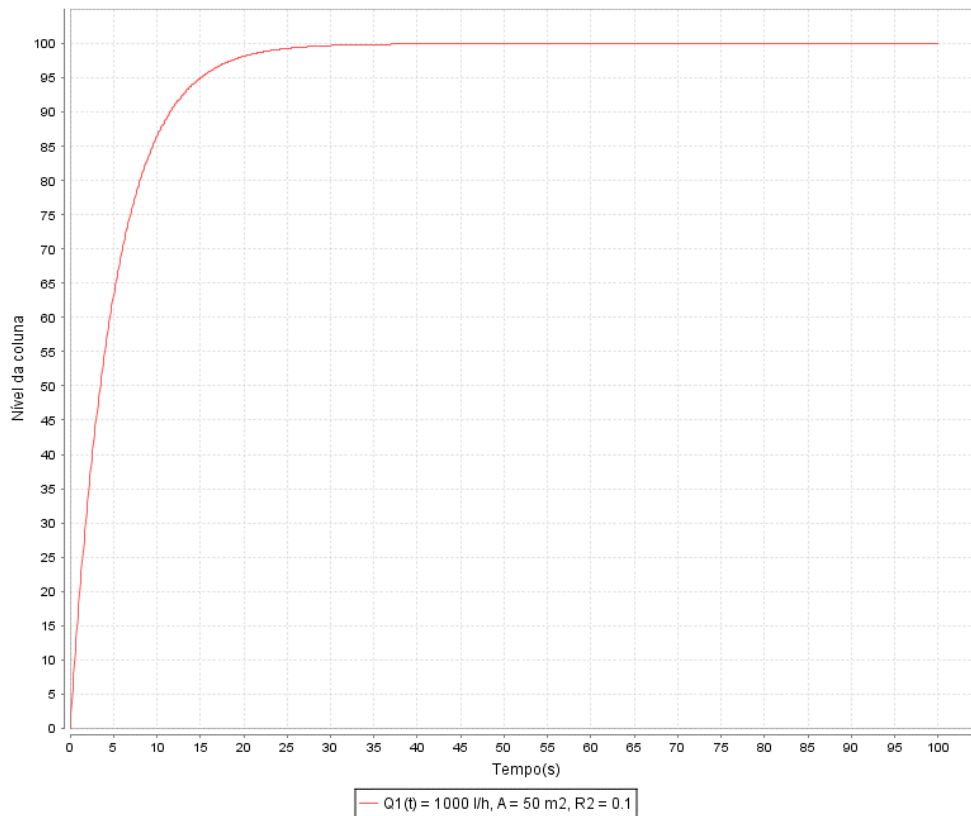
Essa equação é a solução desejada para a equação de transferência (equação 7), com a qual pode-se traçar o gráfico do problema real referido acima. De posse da equação 8, podemos atribuir valores a  $t$  e determinar os valores da função  $h(t)$ , conforme mostra a Tabela 1.

**Tabela 1** - Variação de  $h$  com o tempo.

t	0	5	10	$\infty$
h(t)	0	63.212055882855765	86.46647167633873	100

Temos, então, condições de gerar o gráfico **h(t) versus t** mostrado na Figura 4.

**Simulação de um tanque de água**



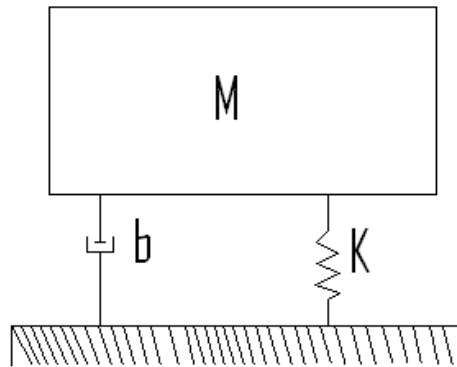
**Figura 4** - Representação gráfica da solução da equação 3.6 para o primeiro problema.

A equação 8, no entanto, não permite associar valores diferentes à vazão de entrada e nem à resistência de saída. Assumindo que  $q_1(t)$  e  $R_2$  são constantes, podemos resolver esta deficiência através da equação 9:

$$h(t) = q_1(t) * R_2 - q_1(t) * R_2 * e^{-t/A \cdot R_2} \quad (9)$$

Essa é a equação com parâmetros dinâmicos ( $q_1$  e  $R_2$ ), com os quais pode-se alterar os valores e observar as variações na simulação e nos gráficos resultantes, gráficos estes iguais ao da Figura 4. Os alunos devem fazer o mesmo passo a passo para solucionar um problema de modelagem matemática. Além da solução matemática, devem ser capazes de entender o que esse problema representa em uma situação real.

O segundo problema real simulado consistia de outra situação típica de modelagem matemática: o sistema massa-mola-amortecedor. Considerando um sistema que consiste de uma massa ( $M$ ), chassi do carro, ligado ao solo por uma mola ( $K$ ) e por um amortecedor ( $b$ ). Essa situação está representado na Figura 5.



**Figura 5** - Representação esquemática do segundo problema real.

Seguindo a resolução igual ao primeiro problema real, obtêm-se na equação 10 a resposta temporal para o deslocamento. Essa equação é a solução do problema matemático e é passível de ser simulada.

$$x_{o(t)} = 1 + \frac{1}{\sqrt{1 - \xi^2}} e^{-\xi \omega t} \text{sen}(\omega \sqrt{1 - \xi^2} t - \text{tg}^{-1} \frac{\sqrt{1 - \xi^2}}{\xi}) \quad (10)$$

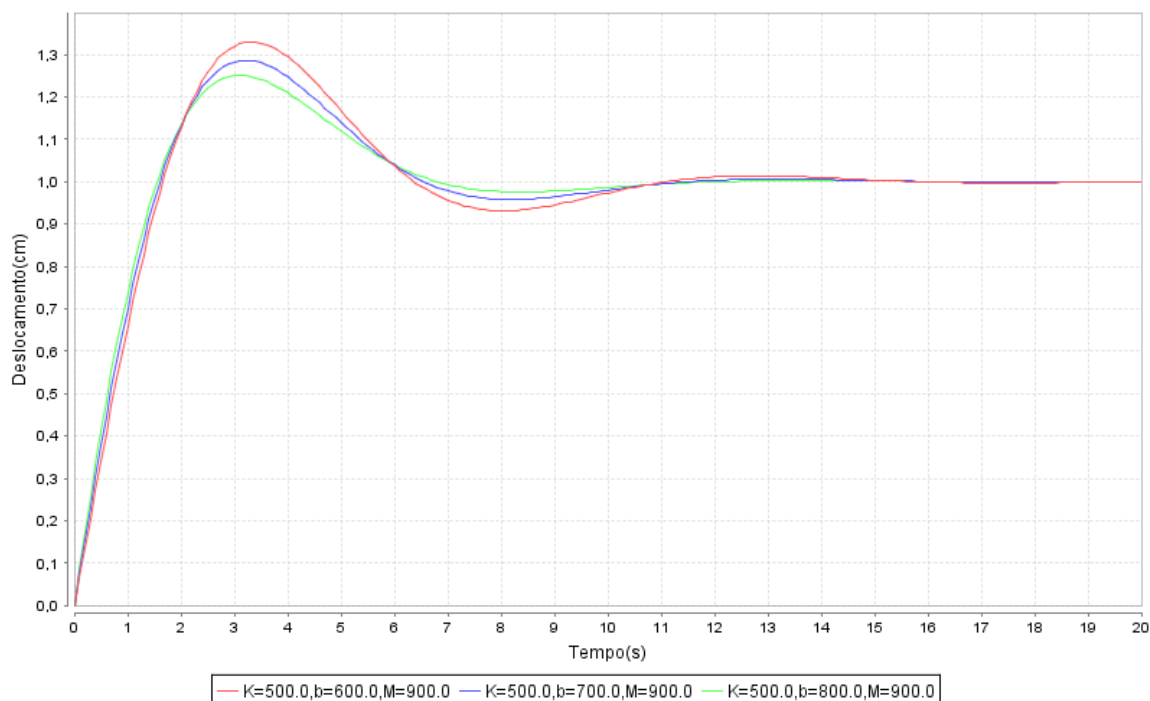
As variáveis dinâmicas a serem simuladas são: o coeficiente de amortecimento ( $\xi$ ) e a frequência natural não amortecida ( $\omega$ ), as equações 11 e 12 mostram o que cada uma dessas variáveis representa.

$$\xi = \frac{b}{M} \cdot \frac{1}{2} \sqrt{\frac{K}{M}} \quad (11)$$

$$\omega = \sqrt{\frac{K}{M}} \quad (12)$$

Além da equação gerou-se um gráfico conforme Figura 6, representando a situação simulada:

### Simulação massa-mola-amortecedor



**Figura 6** - Representação gráfica da solução da equação 4 para o primeiro problema.

A solução desses dois problemas reais é a base para construção da simulação. Feito esse primeiro passo, partiu-se para o desenvolvimento do *software* visando esses problemas.

### 3.2 Planejamento do *software*

A construção dos componentes do *software* demandou estabelecer, previamente, a linguagem de programação a ser utilizada. A escolha recaiu sobre a linguagem Java e C++.

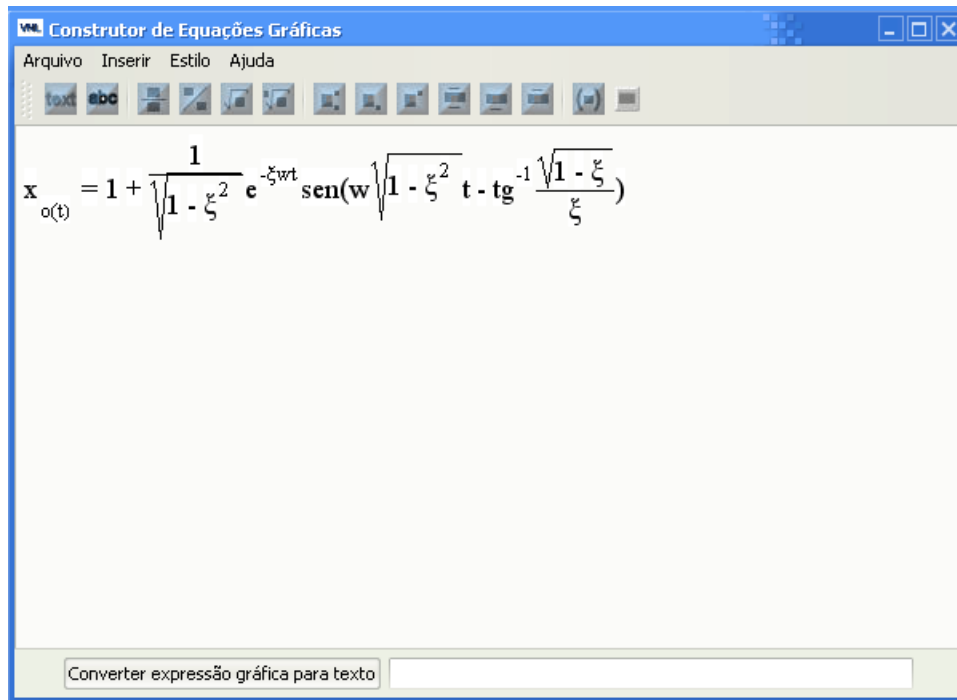
Java é uma linguagem de programação multi-plataforma e orientada a objetos. Os *softwares* desenvolvidos por esta linguagem podem ser distribuídos via Internet através de *Applets* ou executados em qualquer sistema operacional utilizando o JVM específico daquele sistema (DEITEL,2005). C++ é, também, uma linguagem de programação, multi-plataforma e orientada a objetos. Acontece que no quesito, simulação 3D, essa linguagem oferece maior suporte e é mais eficiente. Apenas o componente de exibição de animações 3D foi desenvolvido na linguagem C++ junto com o *game engine* Ogre3D (OGRE, 2006).

Ogre3D é um motor de jogo robusto, gratuito e com uma ampla comunidade de desenvolvedores, que constantemente atualizam as funcionalidades do mesmo, para esse projeto as funcionalidades de simulação 3D são suficientes.

### 3.3 Desenvolvimento do *software*

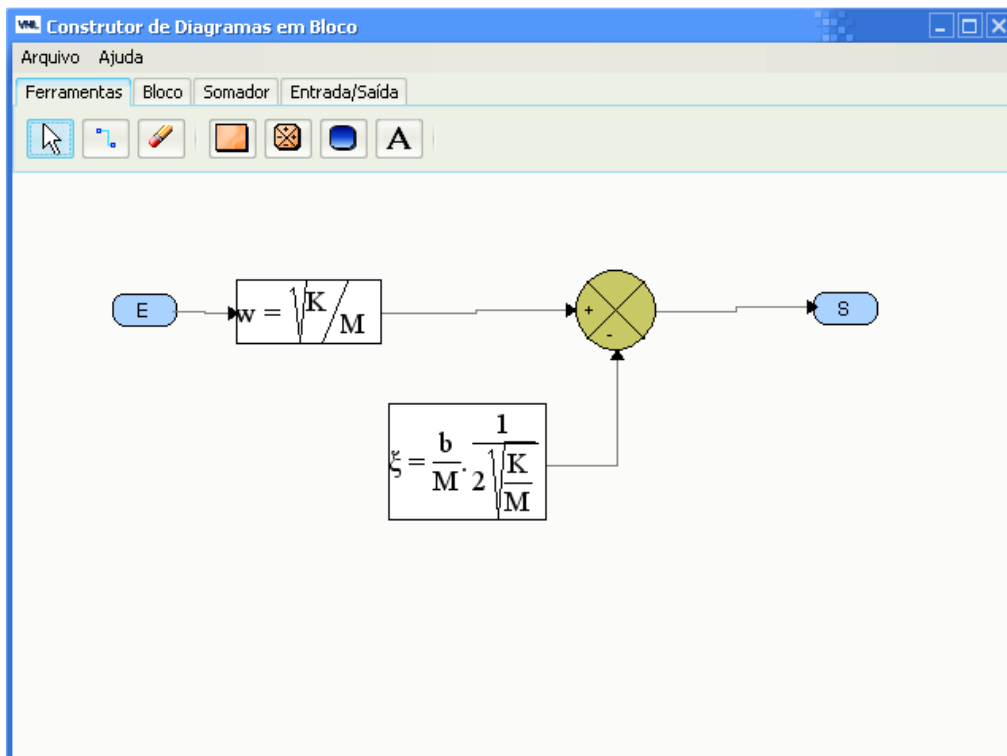
O primeiro componente a ser desenvolvido foi o construtor de equações gráficas. Para essa tarefa utilizou-se como base o *software* *Mathador* (SANTUCCI,2005). Esse *software* é um construtor de equações gráficas desenvolvido em Java e de código aberto. Para adequar-se a esse trabalho novas funcionalidades foram implementadas. A primeira das modificações foi a alteração do código fonte do *parser* do MathML. Com isto, foi possível gerar equações de texto, as quais são interpretadas pelo componente denominado interpretador de equações. Além desta modificação, foram inseridas outras funcionalidades e uma reestruturação da interface gráfica. A Figura 7 demonstra esse componente e uma equação criada pelo mesmo.





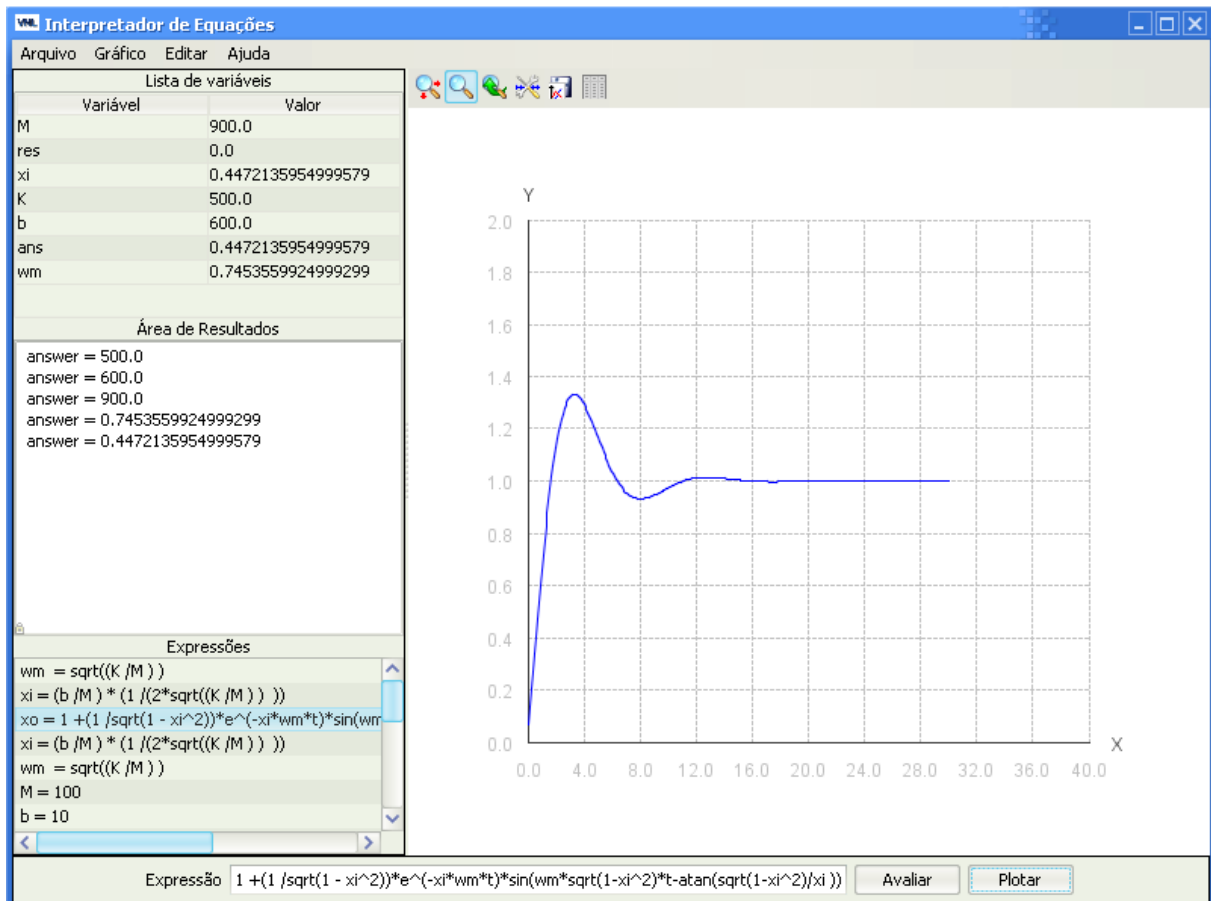
**Figura 7** – Interface gráfica do construtor de equações gráficas.

No desenvolvimento do construtor de diagrama em blocos partiu-se do zero na sua produção, isto é, criaram-se códigos fontes partindo, por exemplo, de bibliotecas gráficas Java, denominadas de componentes *Swing*, existentes. O construtor (Figura 8) é composto de um painel de desenho e de uma barra de ferramentas, os quais viabilizam a construção dos diagramas em bloco. Os objetos de um diagrama em bloco são: bloco, somador/comparador, entrada/saída, conector e texto.



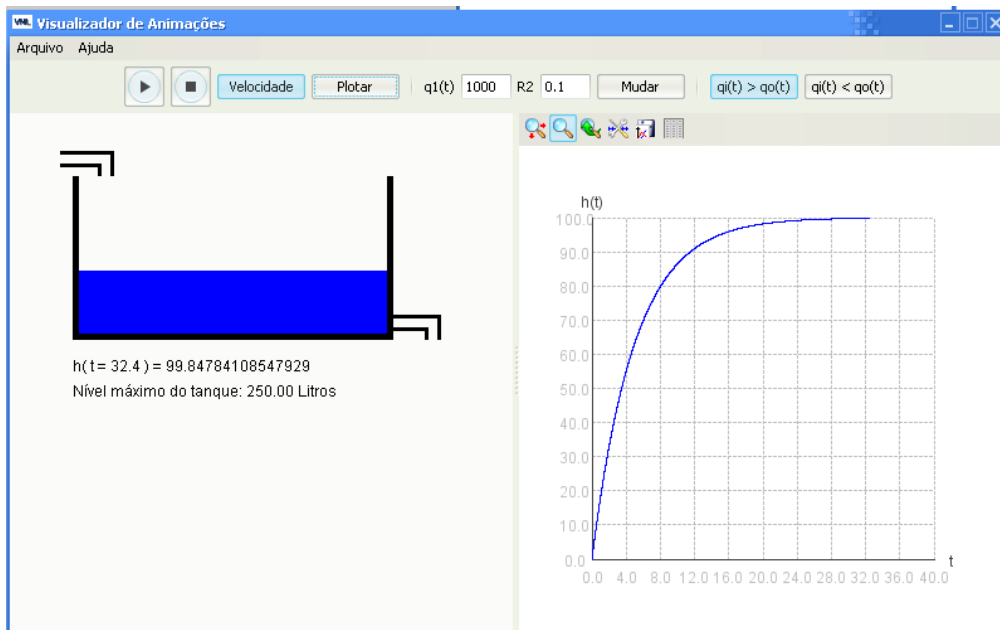
**Figura 8** – Interface gráfica do construtor de equações gráficas.

O terceiro componente, o interpretador de equações, permite que as equações criadas pelo usuário sejam solucionadas. Uma conversão da linguagem *MathML* foi feita, na qual as equações são elaboradas no primeiro componente do *software*, para uma linguagem de notação de equação interpretável pelo *software* JEP (JEP,2006). O JEP possui uma vasta gama de funções, desde funções trigonométricas, operadores booleanos e números complexos. Além de interpretar as equações, este componente permite plotar um gráfico destas mesmas equações. Para isto, incorporou-se o *software* *JMathPlot*, o qual é, também, uma aplicação *Java*. O *JMathPlot* permite plotar gráficos em barra, linha, em 3D e gráficos. Para a simulação dos dois problemas reais foi necessário apenas este tipo de gráfico de linhas. Esse *software* pode ser utilizado em uma aula presencial podendo auxiliar o professor ao resolver equações matemáticas e exibi-las com o auxílio de um computador e um *datashow*. Esse tipo de aula tende a prender a atenção dos alunos e agiliza o processo de resolução de problemas matemáticos em sala de aula. Além de uma ferramenta para auxílio na apresentação de aulas, essa ferramenta pode ser utilizada pelo aluno para solucionar diversos problemas. Essa ferramenta também pode ser adaptada para o ensino de matemática do ensino médio e outras funções menos complexas. A Figura 9 demonstra a interface gráfica desse *software*.



**Figura 9** – Interface gráfica do interpretador de equações.

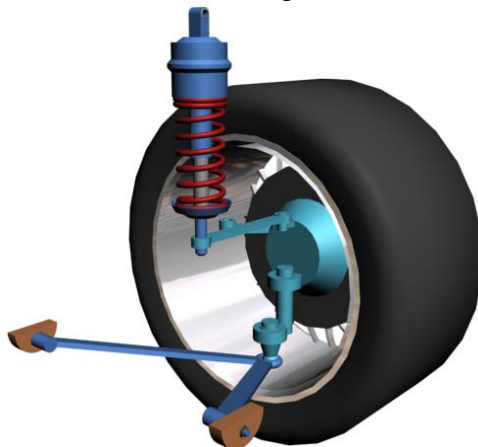
O quarto componente, o visualizador de animações pré-definidas, requereu selecionar um sistema físico e solucioná-lo. A partir do sistema físico criou-se um modelo computacional em *Java*, para representar a realidade, ou seja, um código-fonte foi criado. A Figura 10 exhibe o primeiro problema simulado.



**Figura 10** – Interface gráfica do visualizador de animações 2D.

O primeiro problema real foi simulado em 2D, desenvolvido em Java. Da equação representante do modelo físico, foram selecionadas variáveis da equação passíveis de serem alteradas e/ou que tivessem impacto no resultado final da simulação. A idéia básica do desenvolvimento, aqui, é permitir ao usuário arbitrar os valores com os valores das variáveis e perceber o impacto no resultado final da modelagem. Este tipo de abordagem é defendido por Morrison, (MORRISON & IP,2001) e é chamado de *Rule-based Simulation* (RBS). A simulação baseada em regras consiste em simular modelos de sistemas reais apresentando para o usuário um conjunto de valores de entrada, permitindo a ele alterá-los e analisar os valores de saída. Os valores de saída podem ser entendidos como o nível da água no tanque, a vazão de saída de água no tanque e também a plotagem do gráfico que está em função do tempo e do nível da coluna de água. A importância de uma abordagem RBS está em permitir ao usuário experimentar diferentes cenários de um modelo real, em um ambiente seguro e econômico.

O segundo problema real foi simulado em 3D utilizando-se C++ e Ogre. Para a simulação em 3D foi necessário a criação de um modelo 3D de um sistema de suspensão de um carro. A Figura 11 exemplifica o modelo do sistema de suspensão de um carro.



**Figura 11** – Modelo 3D do sistema de suspensão.

Esse sistema de suspensão é válido para as quatro rodas do carro, com algumas diferenças na construção do *frame* do carro. A Figura 12 exemplifica um carro com seu *frame* e as 4 rodas.



**Figura 12** – Carro 3D e o sistema de suspensão.

Essa estrutura é uma estrutura McPherson básica de um braço tipo A apenas. O aluno deve ser capaz de associar o gráfico da equação plotada com a situação esperada no caso de uma passagem do carro em um quebra-molas, de uma queda da roda em um buraco, da aceleração inicial e de uma freada no carro. Todas essas situações estão simuladas e podem ser apresentadas em tempo real para os alunos através de um computador e um *datashow*. Essas simulações estão disponibilizadas em vídeos para o aluno acessar remotamente através da internet.

### **3.4 Geração de um sistema de ajuda**

Um sistema de ajuda é essencial para auxiliar o usuário no entendimento operacional de uma determinada ferramenta. Deste modo, criou-se um sistema de ajuda na forma de um texto escrito num editor Word. Na criação do texto houve a necessidade imaginar os diversos cenários de dificuldades do usuário ao usar o *software*. Depois de criar o texto do sistema de ajuda, utilizou-se o programa HelpNDoc para transformá-lo num sistema de ajuda do tipo diretório. Sistemas de ajuda do tipo diretório são comuns em todos os *softwares* desenvolvidos para o Microsoft Windows. Criou-se, também, usando o HelpNDoc, um sistema de ajuda *online* para permitir ao usuário acessá-lo em um site da internet. Adicionalmente, foi criado, uma versão em PDF deste sistema de ajuda, utilizando o programa OpenOffice 2.0 que serve para ser impresso e/ou lido na tela do computador pelo usuário. Além do sistema de ajuda foram criados vídeos tutoriais que ensinam a utilizar o *software*.

## **4 CONSIDERAÇÕES FINAIS**

Neste trabalho, efetivamente, desenvolveu-se um *software* educacional baseado no conceito de objetos de aprendizagem aplicado a modelos matemáticos.

O *software* educacional é composto por quatro componentes e um sistema de ajuda. O primeiro componente, o Construtor de Equações Gráficas, permite transformar equações construídas graficamente em equações de texto passíveis de serem interpretadas. Esta funcionalidade foi atingida quando criou-se um código fonte que altera o *parser MathML* de um *software* já existente (*software Mathador*). Digitalmente, este componente é representado

por um painel de desenho e uma barra de ferramentas, com os quais o aluno pode criar equações gráficas, adicionar componentes de uma equação e inserir equações no diagrama em blocos. O segundo componente, o Construtor de Diagramas em Bloco, permite a criação de diagramas em bloco. Esta funcionalidade foi atingida a partir da criação de códigos fontes originais. Digitalmente, este componente é representado por um painel de desenho e uma barra de ferramentas, no mesmo estilo do primeiro componente, os quais permitem ao usuário criar seus diagramas em bloco com o simples clique do *mouse*. O terceiro componente, o Interpretador de Equações, permite interpretar tanto as equações matemáticas construídas no primeiro componente quanto aquelas que o usuário digitar seguindo uma notação padrão. Permite, também, plotar gráficos dessas equações. Estas funcionalidades foram atingidas através da criação de códigos fontes conectores de dois *softwares* já existentes, o JEP e o JMathPlot. O quarto componente, o Visualizador de Animações, permite a visualização de animações previamente elaboradas. Esta funcionalidade foi alcançada através da definição de um problema real e de sua representação por um modelo matemático apropriado, o qual foi simulado por um modelo computacional, viabilizado por um código fonte que foi criado. Para facilitar a exploração da potencialidade destes componentes pelo aluno, criamos um Sistema de Ajuda, descritor do passo a passo a utilização destes componentes.

Os produtos reais gerados por este trabalho estão traduzidos no formato de artigos, de um texto de monografia e de um CD contendo o *software* educacional. Além disto, sugere-se uma nova oportunidade de trabalho futuro, afim de realizar uma “*pesquisa e desenvolvimento de um sistema de avaliação usando objetos de aprendizagem integrados com a tecnologia de jogos virtuais*”.

#### ***Agradecimentos***

Agradecimentos ao Fundo de Incentivo a Pesquisa da PUC Minas pela oportunidade de realizar esse projeto e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq - pela aprovação do projeto de ir complementar esta proposta, através do edital Pibic/CNPq em 08/2006.

## **5 REFERÊNCIAS BIBLIOGRÁFICAS**

DEITEL, H. M., DEITEL P. J., SANTRY, S. E., **Advanced Java 2 Platform: How To Program**, Ed. Prentice Hall, 2003

IP, A., MORRISON, I., Learning Objects In Different Pedagogical Paradigms **Anais...** The 18th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education, 2001

JEP Disponível em <<http://www.singularsys.com/jep/>> Acesso em: 2 de jan. 2006

LALEUF, J. R., SPALTER, A. M. **A Component Repository for Learning Objects**. In: INTERNATIONAL CONFERENCE ON DIGITAL LIBRARIES, ROANOKE, VIRGINIA, January, 2001. **Anais...** 1st ACM/IEEE-CS JOINT CONFERENCE ON DIGITAL LIBRARIES, p. 33-40, 2001

LIMA, C. L., LUCHESSI, E. M., SILVA, P. A., JARDIM, R. F., BASSO, M. V. A., **Ensino de geometria plana através de mosaicos com a ajuda de um objeto de aprendizagem no padrão RIVED**, Revista Novas Tecnologias na Educação, V.3 N°1, maio, 2005

MAEDA, V. A., SANCHES, T. S., SOUZA, G. M., TAVARES, W. N., FILHO, H. F., RODRIGUES, M., QUINTANILHA, J. A., Desenvolvimento de objetos de aprendizagem para o ensino a distância de geoprocessamento, **Anais... XII Simpósio Brasileiro de Sensoriamento Remoto**, 16-21, 2005

MASETTO, M. T., **Competência Pedagógica do Professor Universitário**, Summus Editorial, 2003.

OGRE, *Game Engine Ogre3D* Disponível em <<http://www.ogre3d.org>> Acesso em: 1 de ago. 2006

RIVED. Rede Internacional Virtual de Educação. Disponível em: <<http://rived.proinfo.mec.gov.br/>> Acesso em: 1 de mai. de 2005.

SANTUCCI, V. *Software Mathador*. Disponível em <<http://www.dipmat.unipg.it/~milani/mathador/index.html>> Acesso em: 1 de dez. de 2005

SILVA, A. P., OLIVEIRA H. D., FRÉRE, A. F., **Desenvolvimento de Jogo Computadorizado para Auxiliar o Letramento de Crianças com Hiperatividade Via Internet**, Universidade de Mogi Das Cruzes/Núcleo de Pesquisas Tecnológicas – abril de 2005

SILVEIRA, S. R., BARONE, D. A. C., **Jogos Educativos computadorizados utilizando a abordagem de algoritmos genéticos**, IV Congresso Ribie, 1998

VYGOTSKY, L. S. **O papel do brinquedo no desenvolvimento**. In: A formação social da mente. São Paulo: Martins Fontes, 1989. 168p. p.106-118.

WILEY, D.A. **The Instructional Use of Learning Objects**: Online Version. Disponível em: <<http://www.reusability.org/read/>>. Acesso em: 1 de mai. de 2005

## **DEVELOPMENT OF A SOFTWARE BASED ON LEARNING OBJECTS APPLIED TO MATH MODELS**

**Abstract:** *In this work, we used the concept of learning objects to develop educational software focused on teaching mathematical modeling. This educational software attends the technical issues of accessibility, reusability, and interoperability. Thus benefit the developer-teacher; allowing him/her to build new learning objects from the developed components of this software. This software is a typical learning object which is built to stimulate the students to be creative and also researchers of new possibilities that will help them solve math problems once considered complexes. With this software, the student will have the joy to prove, virtually, the union between theory and practice which will stimulate him/her to be a self-learner and at the same time, to improve his scientific and technological development. This learning object is formed by four components (Graphical Equation Builder, Block Diagram Builder, Equation Interpreter and Animation Viewer) and a help system. The implementation of these components was possible due to the creation of source codes. The*

*products of this work are materialized in this monographic text and a CD with the our educational software proposed.*

***Key-words:*** *Education, Engineering, Learning Object, Modelling*