



COBENGE 2005

XXXIII - Congresso Brasileiro de Ensino de Engenharia

"Promovendo e valorizando a engenharia em um cenário de constantes mudanças"

12 a 15 de setembro - Campina Grande Pb

Promoção/Organização: ABENGE/UFCG-UFPE

Usando cálculo numérico no Macromedia Flash[®] para simular problemas de Física

Leandro Resende de Pádua, leandrorpadua@yahoo.com.br

ITA–Instituto Tecnológico de Aeronáutica, Departamento de Física – IIEFF

Pç. Marechal Eduardo Gomes, n 50 – Campus do CTA, 12228-900,

São José dos Campos – SP

Fretz Sievers Junior - fretz@comp.ita.br

José Silvério Edmundo Germano, silverio@ita.br

Resumo: A simulação de fenômenos físicos com o auxílio do computador pode ser utilizada com grande eficiência, como ferramenta auxiliar ao professor dentro do ambiente da sala de aula, tanto no ensino da Física básica quanto nos estudos de Física mais avançados. Conseguir fazer o aluno visualizar um fenômeno físico, muitas das vezes complexo, pode se tornar um grande desafio para o professor, que com o auxílio do computador e simulações apropriadas, tem uma alternativa eficaz para contornar essa dificuldade. Porém, desenvolver essas simulações (Objetos de Aprendizagem) explorando a parte gráfica na maioria das vezes não é trivial, devido a complexidade das equações matemáticas que envolvem a maioria dos fenômenos físicos. O que apresentamos nesse projeto, é a criação de algumas simulações de problemas de mecânica, basicamente sistemas oscilatórios, utilizando o programa Macromedia Flash MX 2004[®]. No desenvolvimento do projeto, utilizamos o método de Runge-Kutta para resolução numérica de equações diferenciais que envolvem os problemas oscilatórios. É importante ressaltar que as simulações desenvolvidas, estão sendo utilizadas nos cursos de Física que são ministrados no Instituto Tecnológico de Aeronáutica (ITA) com resultados bastante satisfatórios, despertando um maior interesse e uma melhor compreensão do temas abordados.

Palavras-chave: *Flash, Cálculo numérico, Ensino a Distância, Objetos de aprendizagem.*

I. Introdução

Fazer com que o aluno consiga visualizar no espaço, a representação gráfica do que está sendo ensinado, é uma das maiores dificuldades que os professores de Física enfrentam. Existem algumas formas de minimizar esse problema, sendo uma delas a utilização do computador dentro do ambiente da sala de aula, um recurso que já vem sendo explorado e que mostra resultados promissores.

Nem sempre os alunos das disciplinas de Física Básica dominam os conceitos de primitiva e derivada. A associação desses conceitos com os fenômenos da física é ainda mais rara, e torna-se difícil a construção de relações que envolvam, por exemplo, o movimento oscilatório amortecido e forçado, tais como: constante de elasticidade da mola (k), gráfico de amortecimento em relação ao tempo e velocidade. Esses conceitos, do ponto de vista teórico, são bastante conhecidos e encontram-se descritos na maioria dos livros textos de Física (Tipler, 2000; Halliday, 2002; Serway, 1992). Uma forma eficiente de viabilizar a construção destas relações, é a utilização de simulações de um experimento que mostre todas as relações envolvidas. Neste trabalho, desenvolveremos essas simulações, também conhecidos como objetos de aprendizagem, com o programa Macromedia Flash®. O problema que estamos interessados em ilustrar estão relacionados com os sistemas oscilatórios: Movimento Harmônico Simples, Oscilação Amortecida e Oscilação Forçada que mostrará em tempo real, além das animações os gráficos relacionados ao sistema. Com simulações desse tipo, como ferramenta auxiliar dentro do ambiente da sala de aula, o professor poderá tornar a aula mais atrativa e dinâmica, podendo concentrar – nestas ocasiões - seus esforços nos aspectos conceituais do assunto tratado. Além disso, podem ser feitas perguntas aos alunos que os induzam a pensar e a formular hipóteses, das quais a maior parte pode ser testada no ato. Toda esta atividade é viável mesmo em turmas com número elevado de alunos, utilizando para tanto um computador acoplado a um projetor multimídia, ou em um laboratório de informática onde cada aluno pode ter acesso aos Objetos de Aprendizagem.

A grande vantagem em usar o Macromedia Flash® nesses tipos de aplicações está no alto nível de seus recursos multimídia e principalmente na facilidade de utilização dos mesmos. Originalmente ele foi criado com o objetivo de ser um programa para criação de aplicações *web* e com sua evolução, passou a utilizar uma linguagem de programação, o *ActionScript*, que foi baseado no *JavaScript*.

Um programa que tem em primeiro plano o desenvolvimento de *websites* traz consigo um elevado número de ferramentas para a criação de aplicações gráficas. Em contrapartida, agora nos deparamos com a dificuldade de resolução de problemas de física que apresentam uma matemática complexa, em particular, aqueles que envolvam equações diferenciais. Entretanto, o que vamos mostrar a seguir é como acabar com essa dificuldade de uma maneira simples, utilizando para isso o cálculo numérico.

Neste projeto foi feito um programa com Macromedia Flash® que simula o movimento oscilatório amortecido e forçado, utilizando o método de Runge-Kutta para solução das equações de movimento que envolvem tais problemas.

II. PROBLEMA: SIMULAÇÃO DE OSCILAÇÃO AMORTECIDA.

O Objeto de Aprendizagem que iremos descrever simula uma oscilação de um sistema massa-mola com amortecimento viscoso com uma força periódica externa do tipo $F_o \cos(w_f t)$ aplicada sobre ele. Onde o usuário interage modificando a constante elástica da mola (K), a viscosidade do fluido (C), o parâmetro da força aplicada (F_o), a frequência angular da força (w_f) e impondo uma deformação inicial da mola, bastando para isso, arrastar o bloco com o mouse. Para iniciar a simulação novamente e para o problema voltar às condições iniciais deve ser pressionado o botão “reset”. São considerados os seguintes

parâmetros iniciais: massa = 1Kg , volume = 100 cm³, tamanho da mola=1m, densidade da água=1 g/cm³, gravidade =9,78 m/s². Esse problema pode ser visualizado na figura 1.0.

O sistema massa-mola é representado através de uma mola segurada por uma mão de um aluno, onde a mola se encontra submersa em um líquido cuja viscosidade é definida pelo usuário, de forma que o bloco está preso à mola. Conforme o usuário posiciona o bloco com o mouse, o bloco terá uma nova posição e aplicará uma deformação na mola que após o usuário soltar o botão do mouse, o bloco será puxado ou empurrado (depende da posição do bloco) pela força de elasticidade da mola e o sistema traça um gráfico de um sistema amortecido e forçado em tempo real, até que entre em regime estacionário com oscilações constantes de frequência ω_f . A figura 2.0 mostra o estado inicial da simulação e a figura 3.0 mostra o estado estacionário que aparece após um certo tempo.



Figura 1.0 – Simulação de Oscilação Amortecida e Forçada, posição inicial



Figura 2.0 – Bloco arrastado pelo aluno. Começa a simulação.

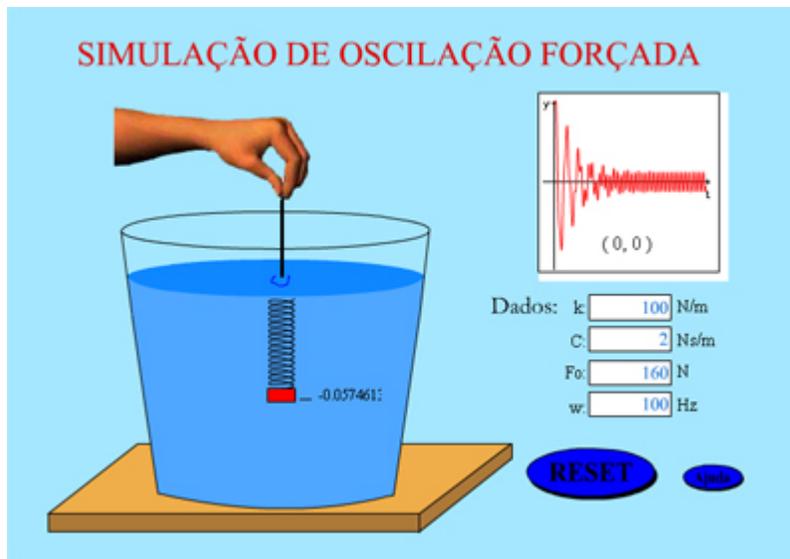


Figura 3.0 –Término da simulação com gráfico - Regime estacionário

III. EQUAÇÕES DIFERENCIAIS DO MOVIMENTO OCILATÓRIO

A equação que dita o movimento oscilatório amortecido e forçado pode ser escrita como:

$$m\ddot{x} + c\dot{x} + kx = F_o \cos(w_f t)$$

Apesar dela possuir solução analítica, vamos utilizar o método numérico de Runge-Kutta para exemplificar como utilizá-lo dentro do Flash.

IV. EQUAÇÕES DO PROBLEMA – O RUNGE-KUTTA.

Os métodos de Runge-Kutta são uma família de métodos numéricos para solucionar equações diferenciais ordinárias. São métodos que podem ser obtidos pela série de Taylor sem a necessidade de calcular qualquer derivada. Para resolução desse problema, utilizamos o modelo clássico método de Runge-Kutta de 4ª ordem. Por sua dedução bastante trabalhosa, limitamo-nos a enunciar apenas sua expressão utilizada.

Esse método consiste em estimar o valor da função em vários pontos intermediários e o valor solução é encontrado pela média ponderada entre esses pontos. Sua escolha para o projeto foi devida à sua grande precisão e pelo fato de poder ser utilizado na resolução de diversos tipos de equações.

A expressão do método de Runge-Kutta de 4ª ordem é dada por:

$$y_{i+1} = y_i + h[\dot{y}_i + \frac{1}{6}(k_1 + k_2 + k_3)]$$

$$\dot{y}_{i+1} = \dot{y}_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

onde

$$k_1 = hf(t_i, x_i, \dot{x}_i)$$

$$k_2 = hf(t_i + \frac{h}{2}, x_i + \frac{h}{2}\dot{x}_i + \frac{k_1}{2})$$

$$k_3 = hf(t_i + \frac{h}{2}, x_i + \frac{h}{2}\dot{x}_i + \frac{h}{4}k_1, \dot{x}_i + \frac{k_2}{2})$$

$$k_4 = hf(t_i + h, x_i + h\dot{x}_i + \frac{h}{2}k_2, \dot{x}_i + k_3)$$

No primeiro quadro, mostrado na figura 1.0, declara-se o passo de indução, que chamamos de h (0,016 foi usado no projeto), juntamente as outras variáveis, que no caso do projeto foram: $kk = k/m$, $cc = c/m$, $ff = F_0/m$, $ww = w_f$. Sendo que os valores de k , c , F_0 e w_f podem ser alterados pelo usuário e “ m ” é a massa do bloco, que foi definida como 1kg, mas também poderia deixá-la a critério do operador. Também consideramos o tamanho da mola como 1m e a quantidade de vezes que é executado o cálculo da posição $n = 500$.

Vincula-se então, a posição e a velocidade do bloco com os valores $ys[n]$ e $vs[n]$, respectivamente, sendo que $vs[n]$ é vinculado indiretamente, pois é utilizado no cálculo de $ys[n]$, o qual é diretamente vinculado à posição vertical do bloco. Observe que tanto $vs[n]$ e $ys[n]$ são matrizes que são utilizadas para realizar a animação do bloco e para traçar o gráfico, assim temos que para cada período de tempo deve ser calculado um novo valor para essas matrizes. É nesse ponto que é necessário utilizar o Runge-Kutta, para conseguir esses valores.

Para realizar os cálculos é preciso criar um objeto do tipo *Movie Clip*, pois este tipo de objeto possui uma linha de tempo independente da linha de tempo principal, tornando possível assim colocar o código do Runge-Kutta (listagem 1.0) em um de seus quadros e realizar sucessivos *loop's* sobre esse quadro sem alterar o principal. Dessa forma, obtemos os valores da posição e da velocidade em cada instante, sendo que o valor de $ys[0]$ é gravado com a imposição da posição inicial do bloco pelo usuário.

De posse do valor e da posição do bloco, basta acrescentar uma ação para que ele se movimente de acordo com esse dado. É interessante notar o fato de uma característica do Flash auxiliar especificamente essa aplicação sobre movimento oscilatório: como a mola é feita utilizando desenho vetorial, a sua deformação durante a animação não ocasiona perda de qualidade gráfica, o que não aconteceria com um *bitmap*. Tal fato garante uma melhor apresentação ao programa, tornando-o mais atrativo.

A seguir, na Listagem 1.0 está o código utilizado no cálculo do Runge-Kutta para o movimento simulado.

```

//Cálculo da Força inicial agindo sobre o bloco.

f = m*g - g*vol*1;    // Peso menos o empuxo

//Nesse ponto impomos uma condição para que o cálculo do Runge-Kutta
seja executado pelo programa (tam é a variável que guarda a amplitude
inicial).

amp = Math.abs(ys[0]);
for(n=0;((amp<tam/100)&&(n<500));n++)
{
    //São declarados os quatro parâmetros do método Rung-Kutta

    k1=h*((ff)*(Math.cos(ww*n)) + f/m -(kk)*ys[n]-(cc)*vs[n]);
    k2=h*((ff)*(Math.cos(ww*(n+h/2)))+ f/m-(kk)*(ys[n]+(h/2)*vs[n])-(
    (cc)*(vs[n]+k1/2)));

    k3=h*((ff)*(Math.cos(ww*(n+h/2)))+f/m -(kk)*(ys[n]+(h/2)*vs[n]+
    (h/4)*k1)-(cc)*(vs[n]+k2/2));
    k4=h*((ff)*(Math.cos(ww*(n+h))) + f/m -(kk)*(ys[n]+h*vs[n]+
    (h/2)*k2v)-(cc)*(vs[n]+k3));
    // Serão guardados os valores da posição e da velocidade dentro de
    duas matrizes (ys e vs).
    ys.push(ys[n]+h*(vs[n]+(1/6)*(k1+k2+k3)));
    vs.push(vs[n] + (1/6)*(k1+2*k2+2*k3+k4));
}

```

Listagem 1.0 – Códigos inseridos em Action Script

V. Usando o Flash

Macromedia Flash MX 2004 é uma ferramenta de desenvolvimento com muitos recursos, possibilitando a entrega de uma variedade de conteúdos dinâmicos na *web*. Uma pesquisa realizada pela Macromedia informa que o *Flash Player* está instalado em 98% dos *browsers* conectados à *web*. Ao contrário de um código HTML estático, uma aplicação feita no Macromedia Flash® pode responder rapidamente sem a necessidade de se fazer algum processamento no servidor. Essa ferramenta atende aos requisitos para os nossos estudos de caso para criação de Objetos de Aprendizagem em física. É importante observar também que a linguagem *ActionScript* é de fácil utilização, tendo em vista que em seu ambiente de desenvolvimento encontra-se um *help* bem completo dessa linguagem.

Para desenvolver o projeto, primeiramente foi criado o ambiente gráfico com o qual iríamos trabalhar. Para isso foram utilizadas as ferramentas de desenho disponíveis no programa, que são bem simples de usar. O Flash, como já citado, tem ainda a vantagem de trabalhar com desenho vetorial, que possui recursos de fácil manipulação, o que em geral, torna o processo de desenho mais simples para o propósito de representação de situações físicas. Lembrando que também é possível trabalhar com outros tipos de gráficos dentro do Flash (jpg, gif, bmp, etc.), além de sons e vídeos em diversos formatos.

Depois de montada toda a parte gráfica foi feita a parte de interação. Para tanto, foram inseridos campos de texto que guardam os valores das constantes utilizadas no

movimento e foi acoplado um *script* ao bloco do sistema massa-mola para que o usuário o arraste, impondo assim uma deformação inicial sobre a mola.

O próximo passo foi criar um novo *Movie Clip* para o gráfico do movimento, que consistia em um eixo de coordenadas com um outro objeto do tipo *Movie Clip* na origem criado com o comando *createEmptyMovieClip*. Esse objeto é uma linha dinâmica, a qual vinculamos a posição com as matrizes $ys[n]$ e $vs[n]$ com o comando *lineTo*. Assim, essa linha é traçada ao mesmo tempo que o bloco se move, pois os dois estão vinculados às mesmas variáveis.

Após isso, adicionou-se o *Movie Clip* com os cálculos do Runge-Kutta. ao filme e o Objeto de Aprendizagem estava pronto. Depois de acabada, a interface do programa ficou como mostra a figura 1.0

VI. Conclusão

Como um dos resultados deste trabalho, podemos citar o fato de que programas criados no Macromedia Flash® estão sendo utilizados nas aulas de física que são ministradas no ITA, tendo uma grande aceitação tanto por parte dos professores (que na maioria das vezes tem dificuldade de incorporar esse tipo de tecnologia no seu dia a dia da sala de aula) quanto pelos alunos. As simulações auxiliam os professores a ilustrar com clareza, e em tempo real, todas as possibilidades envolvidas em tais situações físicas, melhorando de forma concreta a qualidade das informações que são passadas pelos professores, contribuindo assim para o aperfeiçoamento do processo ensino-aprendizagem. Outro ponto importante a ser citado, é que a utilização da técnica baseada no uso do cálculo numérico dentro do Macromedia Flash®, amplia a possibilidade de estudarmos problemas mais complexos dentro do ensino de física, pois com essa técnica podemos resolver as equações que envolvem o problema a ser estudado mesmo que este não tenha solução analítica.

O método de Runge-Kutta facilita na criação dos Objetos de Aprendizagem para a criação dos gráficos, pois não precisamos calcular diferenciais o que dificultaria a implementação do mesmo.

Referências

Fiolhais, Carlos and Trindade, Jorge *Física no computador: o computador como uma ferramenta no ensino e na aprendizagem das ciências físicas*. *Rev. Bras. Ens. Fis.*, Set 2003, vol.25, no.3, p.259-272. ISSN 0102-4744

W. Boyce, *Computers in Physics* **11**, 151 (1997).

W. Boyce, R. DiPrima, *Elementary Differential Equations* **435**, 608 (2001).

Macromedia Flash MX 2004 tutorial oficial
<http://www.macromedia.com/>

Allaire, Jeremy (2002). Macromedia Flash MX – A next –generation rich client.
Macromedia. Available at: <http://www.macromedia.com/software/flash/whitepapers>

Brogan, Pat (1999). Using the Web for Interactive Teaching and Learning. Macromedia. Available at: <http://www.macromedia.com/resource/elearning/whitepapers/>

Jacobsen, P. E-learning Magazine , <http://www.elearningmag.com/elearning/article/articleDetails.jsp=5043>, acesso maio 2002

Knologment Mechanics, http://www.hmedia.com/elearning/kmechanics/k_mechanics3.htm#lcms, acesso setembro 2003

Power elearning, <http://www.power-e.com.br/sistema.htm>, acesso em setembro de 2003.

Reed John A. “Developing Interactive Education Enginner Software for the World Wide Web em Java”, artigo, Computer Education, Vol 30, No ¾ pp. 183-194, 1998.

(Demonstration of the use of Macromedia Flash associated with the numerical calculation in the Physics teaching.)

Using numeric calculation in Macromedia Flash® to simulate problems of Physics

***Abstract:** Simulation of physical phenomena with the computer aid can be used with great efficiency, as auxiliary tool to the teacher inside of the class room, in the Physics teaching, as well as in the studies of Physics more advanced. To get with that the student visualizes the phenomenon physical, a lot of times of difficult understanding, becomes a great challenge for the teacher, that with the computer and appropriate simulations, has an effective alternative to minimize this difficulty. However, the development of simulations (Objects of Learning), exploring the graphic part is not trivial, due to complexity of the mathematical equations that involve the physical phenomena. We presented in this project, to development of simulations of some mechanics problems, based on the oscillatory systems, using the program Macromedia Flash MX 2004®. In the development of this project, we used the method of Runge-Kutta, for numeric resolution of the differential equations of the studied problems. As a result of this project, the simulations are being used in the courses of Physics of ITA with satisfactory results, with a crescent interests and a better understanding of the studied phenomena.*

***Key-words:** Physical Phenomena, Method of Runge-Kutta, Learning Objects.*