

## Implementação de uma rede de computadores para ensino de física

Acquadro, R. V.<sup>1</sup>, Gaspar, C. E.<sup>1</sup>, Raggio, P. M.<sup>1</sup>

<sup>1</sup>Instituto de Física Gleb Wataghin – Universidade Estadual de Campinas (UNICAMP)  
Caixa Postal 6561 – CEP 13084-970 – Campinas – SP – Brasil

racquad@led.unicamp.br, carlosgaspar@yahoo.com, praggio@ifi.unicamp.br

**Resumo.** *O trabalho realizado mostra a viabilidade de software livre em uma rede de computadores para o ensino de física. Foram usados computadores fadados a sucata para revitalizar uma rede e reaproveitá-la para beneficiar os estudantes do instituto de física. O software contruído concomitantemente com a rede também ilustra as inúmeras vantagens na adoção de software livre.*

**Abstract.** *The work shows the viability of the free software in a network for the physics teaching. Obsolete computers were used along a new network to benefit the students of the physics institute. The software developed also shows the advantages in the use of free software.*

### 1. Introdução

Nos últimos anos, o uso de recursos computacionais no ramo da educação tem evoluído de maneira espantosa, tanto na melhoria da qualidade quanto na quantidade de ferramentas. Então, por que não explorar melhor estes recursos, tornando os estudos mais concretos e agradáveis aos estudantes?

Hoje em dia, a computação tem se tornado uma ferramenta imprescindível para o ensino e pesquisa de física, sendo utilizada em aquisições de dados, tratamentos estatísticos, simulações, entre tantas outras aplicações. Mas, como conciliar física e computação em tempos de crise, baixas verbas destinadas a compra de recursos computacionais e altos preços de softwares? Isto é possível através da adoção de softwares de baixo custo, mas principalmente através da adoção de software livre pelas suas inúmeras vantagens (especialmente: custo, facilidade de adaptação/personalização e constante desenvolvimento através da colaboração entre os usuários).

Este trabalho procura mostrar como é possível utilizar software livre em uma rede destinada ao ensino de física. O software simulador desenvolvido e toda a configuração da rede de computadores é um bom exemplo de aplicação computacional de baixo custo voltado para o ensino de física.

É importante ressaltar que durante este trabalho não foi usado em momento algum qualquer software proprietário. Desde o desenvolvimento do software até a escrita desta publicação, foram usadas apenas ferramentas livres e de fácil aquisição na Internet.

### 2. Software

#### 2.1. Considerações e objetivos:

O software simulador foi desenvolvido totalmente a partir do zero, mas inspirado em softwares já existentes. O objetivo do software é oferecer ao aluno de física uma ferramenta de simulação de exercícios que podem ser normalmente encontrados nos livros didáticos [1, 2, 3], auxiliando o entendimento da disciplina. O software também pode ser útil ao educador, por oferecer uma maneira simples, rápida e prática de demonstração dos exemplos estudados em sala de aula.

Para atender a estes objetivos, o software se propõe a resolver um certo número de problemas normalmente encontrados nas disciplinas ministradas durante os cursos de mecânica clássica.

## **2.2. Características e funcionamento**

Em seu atual estágio de desenvolvimento, o software conta com quatro tipos de problemas: queda de corpos, osciladores com molas, lançamento de projéteis e pêndulos, todos, com exceção do módulo de pêndulos, levando em consideração atrito viscoso com o meio. A idéia de modularizar o software torna mais fácil uma possível atualização, aumentando a gama de problemas oferecidos. Existe um corpo principal do software onde cada módulo responsável por uma das simulações é devidamente inserido. O programa principal chama os módulos separadamente, dependendo da escolha do usuário. Desta forma é fácil inserir sub-rotinas para a simulação de outros exercícios de mecânica e até de outras áreas da física.

A linguagem de programação adotada para o desenvolvimento do software foi a linguagem C [4], uma das mais utilizadas junto a comunidade acadêmico/científica. Isso foi feito pensando que, no futuro, outros usuários possam realizar alterações ou implementações do programa com certa facilidade. Outra vantagem da linguagem C é sua portabilidade para outros sistemas operacionais. Inicialmente, ele foi desenvolvido para Linux, mas poderá ser compilado em qualquer outro sistema operacional que suporte linguagem C padrão ANSI.

O software é todo executado em modo texto, mas para que o usuário possa ver os gráficos gerados, é necessário que ele possua acesso a alguma interface gráfica (seja ele via SVGALib ou via X-Window). Se o usuário não tiver acesso a este tipo de recurso, o software apenas apresentará os resultados numéricos. Mas, mesmo sem acesso a interfaces gráficas, ainda existe a possibilidade de salvar os gráficos em um arquivo de formato PNG ou PostScript. A capacidade de salvar gráficos em arquivos é extremamente útil quando deseja-se, por exemplo, ilustrar um relatório ou uma publicação. O formato PNG é suportado por uma gama grande de editores de texto, livre e não livres.

Para apresentar o resultado gráfico do problema, o software gera um script que é lido pelo GNUPlot[5]. Neste script estão contidas todas as informações necessárias para a construção do gráfico desejado. O software GNUPlot foi escolhido pois é de fácil programação, pode ser encontrado em qualquer distribuição Linux e é livre.

A interface do software é a mais simples possível. As opções são acessadas a partir de menus, onde cada opção chama o módulo respectivo do software, onde o usuário deverá entrar com dados específicos para o problema. Após a entrada dos dados o software se encarrega de apresentar soluções numéricas e gráficas.

### 2.3. Algoritmos empregados

Em geral, os problemas de mecânica clássica são descritos por equações diferenciais de segunda ordem; porém, em nem todos eles é possível chegar a uma solução analítica. Nesses casos é necessário utilizar métodos numéricos para encontrar as soluções. Dentre os quatro módulos propostos, existiram duas situações em que foram necessários o desenvolvimento de sub-rotinas especiais para a resolução das equações.

Um desses problemas encontrados foi o cálculo do alcance de um projétil com resistência com o ar; este alcance é dado pela raiz de uma função transcendental. O método adotado para este caso foi o método de Newton para encontrar raízes de uma função. A sub-rotina escrita retorna, depois de um número fixo de iterações, o valor do alcance.

Já para o cálculo do período de oscilação do pêndulo, normalmente usa-se a aproximação de  $\sin \theta \approx \theta$ , quando trata-se de ângulos muito pequenos. Foi escrita uma sub-rotina que utiliza método numérico para calcular o período de oscilação em pêndulos para qualquer elongação. Analiticamente, utiliza-se métodos de integrais elípticas, mas estas podem ser transformadas em polinômios, simples de serem calculados [6].

### 2.4. Histórico

Desde a versão 0.4 do simulador (lançada no dia 18 de Setembro de 2002) o programa esteve disponível na Internet para que qualquer interessado pudesse baixar o código fonte. Graças a isso, o programa recebeu grande colaboração de Daniel Ome, professor de segundo grau que, mesmo geograficamente distante (morando na Argentina), esteve muito presente e ativo durante o desenvolvimento, colaborando muito com a resolução de bugs de programação. Isso só foi possível porque o software foi licenciado sob os termos da GPL, tornando-o totalmente livre. A partir da versão 1.0 (atual versão disponível), além do código fonte, os arquivos executáveis compilados para Linux também estão disponíveis para download no seguinte endereço: <http://sourceforge.net/projects/mecanica>.

Ao término do desenvolvimento, foi escrito um Guia do Usuário, para que os usuários possam ter uma orientação sobre o uso do programa. Este guia de referência explica como o problema é simulado, quais as equações envolvidas nos cálculos e quais os resultados retornados ao usuário pelo software e também pode ser encontrado no mesmo endereço do software.

## 3. Rede de computadores

Paralelamente com o desenvolvimento do software, uma rede experimental composta por três PCs IBM 486 foi montada, para que pudessem abrigar software úteis para o uso dos estudantes de física. Atualmente, a rede encontra-se montada, mas apenas como demonstrativo de como é possível utilizar software livre e micros antigos para uso de alunos de um modo geral, atendendo de maneira satisfatória às suas necessidades.

### 3.1. Características da rede

Apesar da obsolescência dos PCs 486, foi possível, graças ao software livre, demonstrar como pode-se revigorar e utilizar de maneira satisfatória esse tipo de máquina, fadada ao esquecimento.

Com um pequeno custo de aquisição de um novo computador, é possível concentrar todo o processamento dos aplicativos neste servidor, e distribuir apenas as informações já processadas para todos os clientes. Dessa maneira, é possível ter o desempenho de um micro de ponta em clientes obsoletos.

Inicialmente instalamos alguns softwares nos clientes, como AbiWord, Gnumeric, Galeon, entre outros. Mas com a idéia de um servidor ligado na rede, pode-se instalar outros softwares mais modernos e completos, como Mozilla, Evolution, OpenOffice, entre tantos outros.

## 4. Resultados alcançados

Apesar da versão final do software ter sido lançada dia 22 de Novembro de 2002, há pouco mais de 6 meses, já foram colhidos alguns resultados muito importantes:

- Participação ativa de um professor de segundo grau argentino no desenvolvimento do programa;
- Criação de um Grupo de Usuários de GNU/Linux da Física (GULF);
- Continuação do projeto por outro aluno de graduação.

## 5. Projetos para o futuro

- Desenvolvimento de um front-end para tornar o software mais amigável para o usuário. Este front-end poderia ser ainda em modo texto (usando a biblioteca *ncurses*, por exemplo) ou gráfico (usando qualquer outra biblioteca, como *QT*, *GTK*);
- Criação de novas frentes. Na verdade este projeto já foi iniciado por Marcelo Rigon, que está desenvolvendo módulos de simulação de problemas ópticos. A idéia é que o software cresça e, cresça e, com isso, tenhamos um módulo mais geral de Física, por exemplo. Neste caso, os módulos Mecânica e Óptica seriam sub-módulos do Física.

## 6. Referências

1. J. B. Marion & S. T. Thornton, *Classical Dynamics of particles and systems*, 4a. Edição, Saunders College Publishing, Philadelphia, 1995.
2. K. R. Symon, *Mecânica*, 5a. Edição, Editora Campus, Rio de Janeiro, 1982.
3. H. M. Nussenzveig, *Física Básica*, 1a. Edição, Editora Edgar Blücher LTDA., São Paulo, 1981.
4. R. Johnsonbaugh & M. Kalin, *C for Scientists and Engineers*, Prentice Hall, New Jersey, 1997
5. T. Williams, & C. Kelley, *Gnuplot Reference Manual*, <http://pi3.informatik.uni-mannheim.de/staff/mitarbeiter/westmann/gnuplot.html>, 1993.
6. M. Abramowitz & I. A. Stegun, *Handbook of mathematical functions*, 1a. Edição, Dover Publications, New York, 1970