



ANÁLISE COMPARATIVA DO USO DO SOFTWARE PROPRIETÁRIO VERSUS SOFTWARE LIVRE NOS CURSOS DE GRADUAÇÃO EM ENGENHARIA

Tatiana de S. Lima – tattydesouza7@gmail.com

Geraldo F. Neto – geraldofurtado@gmail.com

Lindolpho O. de Araújo Júnior – lindolpho@leopoldina.cefetmg.br

Centro Federal de Educação Tecnológica de Minas Gerais – Campus Leopoldina

Rua José Peres, 558 – Centro

36700-000 – Leopoldina–Minas Gerais

Resumo: A utilização de softwares de computação numérica vem se expandindo cada vez mais no ramo da engenharia. Estes softwares são práticos e rápidos na resolução de problemas complexos, que são impossíveis ou extremamente dispendiosos para serem solucionados por seres humanos. O alto custo na aquisição dos softwares proprietários vem dificultando sua utilização. Uma solução a este problema é a utilização dos softwares livres. O software livre permite aos usuários executarem, copiarem, distribuírem, estudarem, modificarem e aperfeiçoarem o mesmo, podendo adaptá-lo às suas necessidades. Na maioria das vezes o software livre é gratuito. Já o software proprietário exige ao usuário que compre uma licença, sendo que há restrições por conta do criador na mudança de seu código fonte. Os softwares de computação numérica mais utilizados nos cursos engenharia são o MATLAB® e o Scilab, sendo um de licença paga e o outro de licença livre, respectivamente. Ambos possuem praticamente as mesmas funções básicas, mas há vantagens e desvantagens a serem discutidas entre um e outro. Este artigo tem como objetivo realizar uma comparação entre os softwares, buscando identificar as vantagens e desvantagens de cada um para sua utilização nos cursos de graduação em engenharia.

Palavras-chave: Software livre, Software proprietário, MATLAB®, Scilab.

1. INTRODUÇÃO

É cada vez mais comum a utilização de softwares de computação numérica e simulação em diversas disciplinas dos cursos de engenharia. Estes softwares são utilizados devido a praticidade e rapidez com que os resultados desejados são obtidos ou pela impossibilidade de resolução de problemas complexos com métodos analíticos clássicos. Com a evolução da informática e do hardware de computador, a disseminação destes softwares é crescente e contínua, supercomputadores são capazes de realizar até 5 mil trilhões de cálculos por segundo, o que seria impossível de ser realizado por todos os humanos ao mesmo tempo. Computadores pessoais têm uma capacidade de processamento de dados menor, cerca de 2 bilhões de cálculos por segundo mas, ainda assim, muito maior do que a de qualquer pessoa. Na vida profissional de um engenheiro, a previsão e controle de processos dentro de plantas industriais por esses softwares é capaz de minimizar custos e otimizar tais processos.



No entanto, enquanto o custo do hardware vem sendo reduzido, o alto custo de aquisição e de manutenção de softwares “proprietários”, como o MATLAB®, dificulta a sua utilização quando os recursos financeiros são escassos, o que muitas das vezes é a realidade das universidades do Brasil. Isto vem motivando a ampliação do uso acadêmico de softwares livres.

A definição básica para software livre segundo a *Free Software Foundation* é a seguinte: qualquer programa que pode ser utilizado, estudado, copiado, modificado e redistribuído sem restrições. O software livre não é necessariamente grátis. Já para o software proprietário é: aquele cuja cópia, modificação e redistribuição podem ser restringidas pelo seu criador ou distribuidor.

Os softwares, denominados softwares proprietários, proporcionaram aos usuários diversas aplicações, porém como já citado, o custo é elevado. Por outro lado, segundo (SILVEIRA, 2004), o software livre é uma conquista, um movimento que luta pelo compartilhamento do conhecimento tecnológico, haja vista se orientar, principalmente, para o benefício de seus usuários.

Com o crescente desenvolvimento do ramo da informática e suas tecnologias, nota-se o grande na utilização de softwares livres nas universidades. Estes programas, além de serem uma ferramenta a mais que auxiliam no aprendizado, têm uma importância muito grande ao correlacionar a vida acadêmica do estudante com a vida profissional, permitindo o desenvolvimento e simulação de projetos. Na engenharia um software livre de computação numérico bastante utilizado é o Scilab.

O objetivo deste trabalho é de realizar uma comparação entre um dos principais softwares proprietários utilizados pelos cursos de graduação de engenharia, MATLAB®, e o seu equivalente livre, tendo também seu uso extremamente difundido dentro dos cursos de graduação, o Scilab.

Este artigo está disposto da seguinte forma: na seção dois é apresentada uma comparação entre os softwares livres e os softwares proprietários, na seção três é apresentada uma descrição sobre o MATLAB®, na seção quatro foi feita uma descrição do Scilab, já na seção cinco se encontra uma comparação entre os dois softwares, MATLAB® e Scilab e, por fim, são realizadas as considerações finais acerca deste artigo.

2. COMPARAÇÃO ENTRE SOFTWARE LIVRE E SOFTWARE PROPRIETÁRIO

Um software livre permite aos usuários executarem, copiarem, distribuírem, estudarem, modificarem e aperfeiçoarem o mesmo, podendo adaptá-lo às suas necessidades.

O termo livre não quer dizer que o software seja gratuito, pois algumas versões podem ser pagas. Segundo (PERENS, 1999), “nem todo programa disponível gratuitamente é um software livre, às vezes existem restrições para o uso (como os softwares governamentais; por exemplo, o Receitanet – da Receita Federal) e nem sempre o código fonte está disponível”. Por essa razão, não se deve utilizar o termo software gratuito, e sim software livre.

Há diferenças bem claras e definidas entre um software livre e um software gratuito. O software livre tem grande importância no mundo da tecnologia e da computação, mesmo não sendo necessariamente gratuito, tem seu código-fonte livre e disponível para ser alterado por qualquer um, sem que haja a necessidade de pagamento para fazer tais alterações.

O uso do programa livre na educação é uma alternativa imprescindível a qualquer projeto educacional, tanto no setor público como no privado. Fatores tais como liberdade, custo, flexibilidade são estratégicos para a condução bem sucedida de projetos educacionais mediados por computador. Para o setor educacional, muitas vezes carente de recursos, o



programa livre é uma alternativa viável e que deve ser considerada seriamente (ALMEIDA, 2002).

A grande maioria dos softwares proprietários é distribuída com as licenças de uso, fazendo com que o usuário não compre um software, mas sim a licença para seu uso. O objetivo da End User License Agreements – Eula é restringir os direitos do usuário e proteger o fabricante do software.

A disponibilidade dos códigos-fonte é pré-requisito para que um software seja considerado livre, pois somente tendo acesso a tais informações o usuário poderá estudar ou modificar o software, sem que seja necessário comunicar ao desenvolvedor ou outra entidade qualquer (HEXSEL, 2002). Para a utilização do software proprietário, é necessária a aquisição de licença de uso, o copyright, pois seus direitos autorais e comerciais são preservados – em outras palavras, ele não é livre, nem gratuito. É, simplesmente, concedida ao usuário a permissão de uso sob certas condições, sendo vedada a sua reprodução, instalação múltipla, alteração, cessão, revenda ou redistribuição sem autorização e pagamento adicional (LAMAS, 2005).

A circulação do software livre na Internet ocorreu em 1983, por intermédio de Stallman, para que se pudessem ter estas liberdades à disposição de todos os usuários de computador (GNU, 2009). Ele foi criado em contraposição ao software proprietário, que possui o código-fonte fechado, ou seja, somente um indivíduo ou um grupo especializado da empresa que o desenvolveu tem controle em relação ao seu código e, conseqüentemente, sobre suas funções, melhorias e correções, tornando-o uma espécie de monopólio (SALEH, 2004).

Segundo (DIDIO, 2005) a documentação de softwares livres é pouco esclarecedora, o que dificulta soluções de problemas. Um dos grandes problemas a serem enfrentados na adoção de sistemas de software livre é a necessidade de maior capacitação dos operadores (técnicos que instalam, configuram e mantêm os sistemas em operação).

Com o crescimento do software livre, o número de profissionais qualificados tende a aumentar. Por isso, de acordo com Didio (2005), o custo do suporte e da manutenção dos softwares livres pode acabar saindo maior que a aquisição de um software proprietário, uma vez que existem poucos profissionais que realmente entendam o processo como um todo.

3. MATLAB®

O MATLAB®(do inglês MATrix LABoratory) é um software “proprietário”, ou registrado, interativo de alta performance voltado para o cálculo numérico e gráficos científicos. Seu ponto forte está na manipulação e cálculos matriciais, como por exemplo, resolução de sistema linear, cálculo de autovalores e autovetores, fatoração de matrizes, entre outros. Além disso, muitas funções especializadas já estão internamente implementadas, de modo que em muitos casos não há a necessidade de se construir muitas coisas.

O software foi criado no final dos anos 70 por Clev e Moler, presidente do departamento de ciências da computação da Universidade do Novo México. Imediatamente se espalhou para outras universidades e encontrou um forte uso no âmbito da comunidade matemática aplicada. Jack Little, um engenheiro, conheceu a linguagem MATLAB® durante uma visita feita por Moler à Universidade de Stanford em 1983 e, reconhecendo o seu potencial comercial, juntou-se a Moler e Steve Bangert. Eles reescreveram o MATLAB® em C, em 1984 fundaram a Math Works e prosseguiram no seu desenvolvimento. O software foi adotado pela primeira vez por engenheiros de projeto de controle, a especialidade de Little, e rapidamente se espalhou para outros campos de aplicação.

Este programa foi desenvolvido para que engenheiros e cientistas possam, através de uma linguagem rica e de fácil sintaxe, desenvolver programas que resolvam seus problemas. A

partir da estrutura básica do MATLAB®, diversos pacotes de expansão foram desenvolvidos, de modo a potencializar e otimizar seu uso para algumas aplicações específicas – e que, em geral, demandariam muitas linhas de código.

Esse sistema permite a resolução de muitos problemas numéricos em apenas uma fração de tempo que se gastaria para escrever um programa semelhante em linguagem Fortran, Basic ou C, por exemplo. Além disso, as soluções dos problemas são expressas no MATLAB® quase exatamente como elas são escritas matematicamente.

Em geral o usuário e a funcionalidade do MATLAB® crescem juntos, pois com o tempo, ele começa a escrever suas próprias funções, agregando capacidades específicas. Isto faz com que, aos poucos, o MATLAB® comece a mimetizar, ou até mesmo suplantando, softwares científicos específicos de diversas áreas. Na “Figura 1”, pode ser observada a interface inicial do MATLAB®.

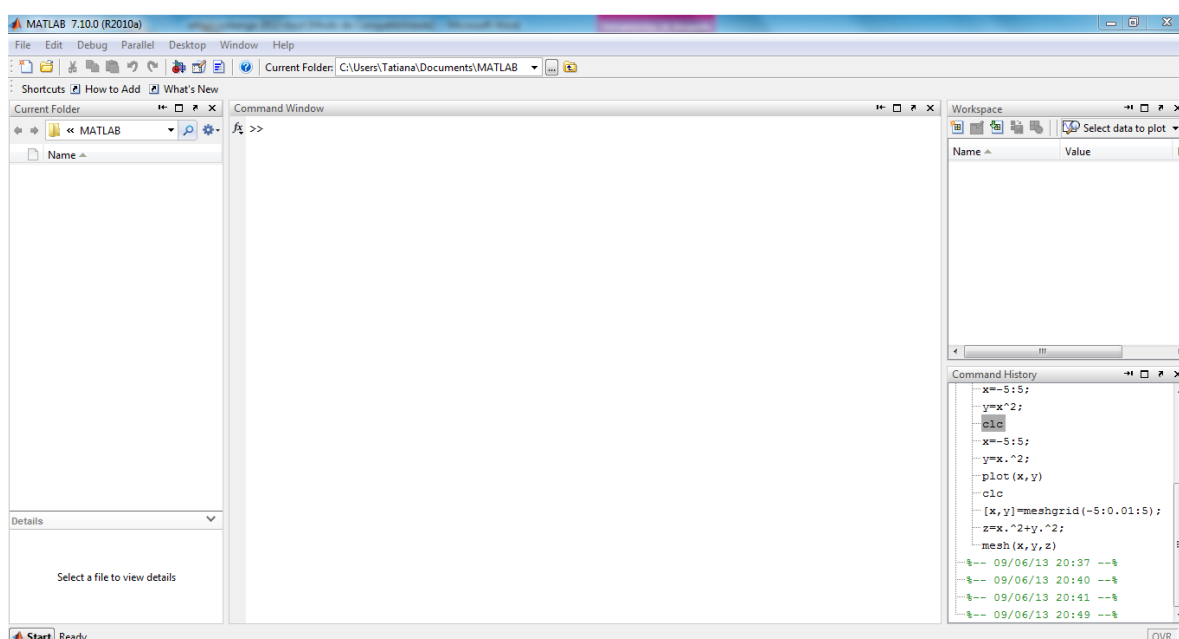


Figura 1 – Interface inicial do MATLAB®

Uma característica interessante é que o MATLAB® é muito mais fácil de aprender do que as linguagens científicas convencionais, tais como C e Fortran. Entretanto, quanto maiores e mais complexas as rotinas e funções da biblioteca, pior fica o seu desempenho quando comparada à sua equivalente em C e/ou Fortran.

Os valores para aquisição desse produto estão em torno de US\$100,00 para versão de estudante e US\$ 3000,00 para versão comercial.

4. SCILAB

Scilab é um software livre e de código aberto para computação numérica proporcionando um ambiente de computação poderosa para aplicações de engenharia e científicos.

Foi desenvolvido desde 1990 por pesquisadores do INRIA e da ENPC na França e mantido pelo Scilab Consortium desde 2003. É distribuído gratuitamente através da Internet (<http://www.scilab.org>) para diversas plataformas, entre estas o Linux. Ele inclui um grande número de bibliotecas (toolboxes) que englobam funções gráficas 2D, 3D e animação,



integração numérica, álgebra linear, interpolação, otimização, modelador de sistemas dinâmicos híbridos e outros. Existem, também, bibliotecas específicas para engenharia, como controle e processamento de sinais, e pode-se expandir o número de funções adicionando-se bibliotecas disponíveis na Internet. O Scilab possui uma linguagem de programação própria que permite a criação de programas numéricos.

Atualmente ele é amplamente utilizado em diversos ambientes industriais e educacionais pelo mundo. Na “Figura 2” é apresentada a interface inicial do Scilab.

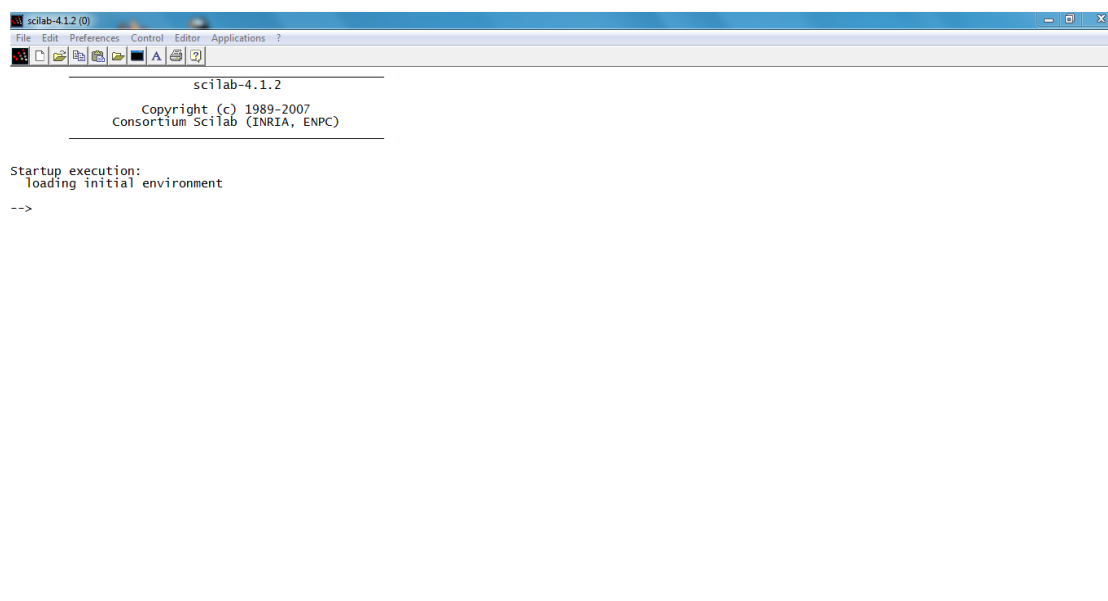


Figura 2 – Interface inicial do Scilab

O programa inclui centenas de funções matemáticas com a possibilidade de adicionar, interativamente, programas de variadas linguagens (C, C++, Fortran, etc.). Ele ainda conta com estruturas sofisticadas de dados, sendo elas: listas, nomes compostos, funções racionais, sistemas lineares, etc..

O Scilab é um ambiente de programação numérica bastante flexível, cujas principais características são (PIRES, 2004): (1) é um programa de distribuição gratuita, com código fonte disponível, sua linguagem é simples e de fácil aprendizado; (2) possui um sistema de auxílio ao usuário (help); (3) apresenta recursos para geração de gráficos bidimensionais, tridimensionais e animações; (4) implementa diversas funções para manipulação de matrizes, incluindo operações de união, acesso e extração de elementos, transposição, adição e multiplicação de matrizes; (5) permite trabalhar com polinômios, funções de transferência, sistemas lineares e grafos; (6) apresenta facilidades para a definição de funções; (7) permite o acesso a rotinas escritas nas linguagens FORTRAN ou C; (8) pode ser acessado por programas de computação simbólica como o Maple, que é um programa comercial, ou o MuPAD, que é um programa livre para uso em instituições de ensino/pesquisa; (9) suporta o desenvolvimento de conjuntos de funções voltadas para aplicações específicas, os chamados toolboxes.

5. COMPARAÇÃO ENTRE OS PRINCIPAIS SOFTWARES DE COMPUTAÇÃO NUMÉRICA: SCILAB E MATLAB®

Fazendo uma comparação do Scilab com seu concorrente, o MATLAB®, nota-se que apesar de possuírem as mesmas funções básicas o Scilab possui uma interface bem menos intuitiva que a do MATLAB®, apresenta um número menor de ferramentas e bibliotecas e não possui todas as funções implementadas.

O MATLAB® apresenta uma infinidade de vantagens em relação ao Scilab. No entanto, para aplicações nos cursos universitários este último se torna viável devido à sua gratuidade.

O Scilab mantém a mesma estrutura básica do MATLAB®, com uma sintaxe praticamente idêntica. Este software aceita programas desenvolvidos em MATLAB®, versatilidade esta que tem sido apontada como uma de suas vantagens. O usuário de Scilab, porém, precisa lidar com uma interface muito menos rica em recursos e não terá à sua disposição a vasta gama de pacotes de utilidade do MATLAB®.

As funções definidas no Scilab não são consideradas arquivo separado como é definido no MATLAB®. Elas constituem variáveis no ambiente Scilab. Uma ou mais funções definidas pelo usuário podem ser definidas em um único arquivo. E o nome do arquivo não necessariamente precisa estar relacionado ao nome da função. As funções não são automaticamente carregadas no Scilab, como são no MATLAB® após serem chamadas através dos seus nomes. Normalmente, deve ser executado o comando `getf` (“function_name”), antes de usar a função. As funções podem também ser definidas on-line (são as denominadas funções internas), através do comando `deff`. Para executar um arquivo com código em Scilab é preciso usar o comando `exec` (“filename”). Já no MATLAB® basta digitar o nome do arquivo e executá-lo.

O Scilab ainda possui alguns bugs e problemas de compatibilidade que já foram superados pelo seu concorrente. Por todas estas razões, parece hoje realista dizer que, em médio prazo, o MATLAB® ainda será a ferramenta mais utilizada.

Para realizar uma comparação gráfica entre os dois softwares, estão apresentados nas “Figuras 3 e 4” gráficos em 2D e 3D, e as respectivas linhas de código utilizadas para a geração dos gráficos no Scilab, já nas “Figuras 5 e 6” estão representados dos gráficos em 2D e 3D no MATLAB®.

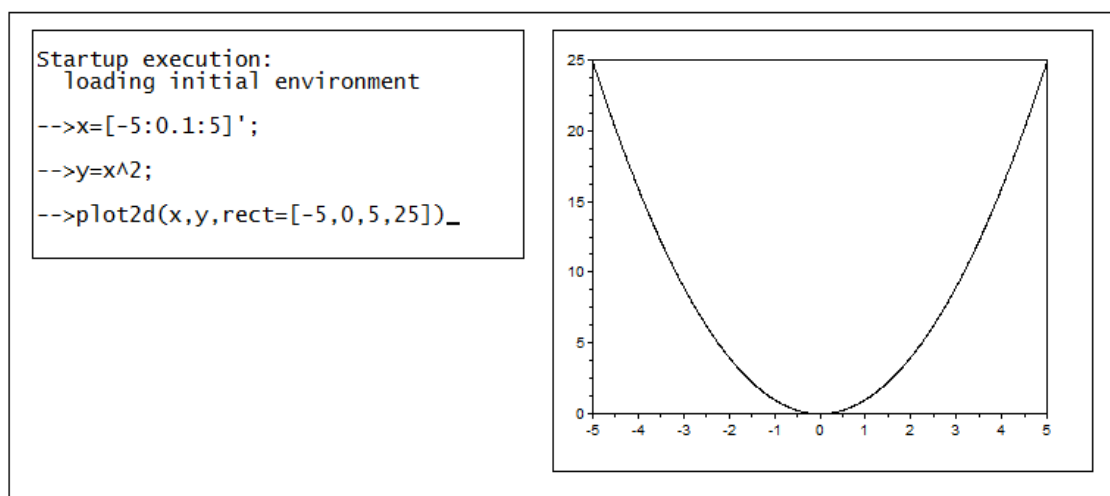


Figura 3 – Representação de um gráfico 2D no Scilab

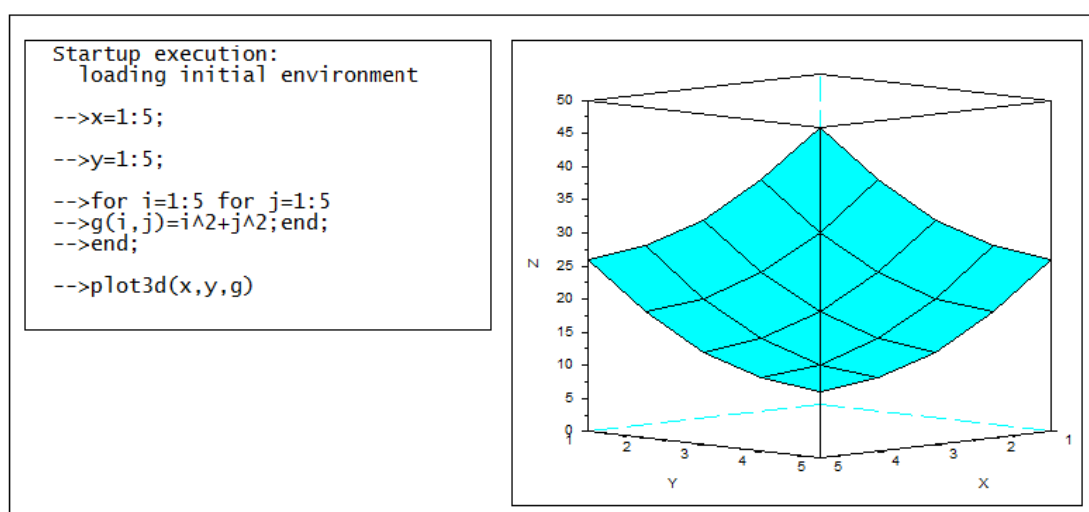


Figura 4 – Representação de um gráfico 3D no Scilab

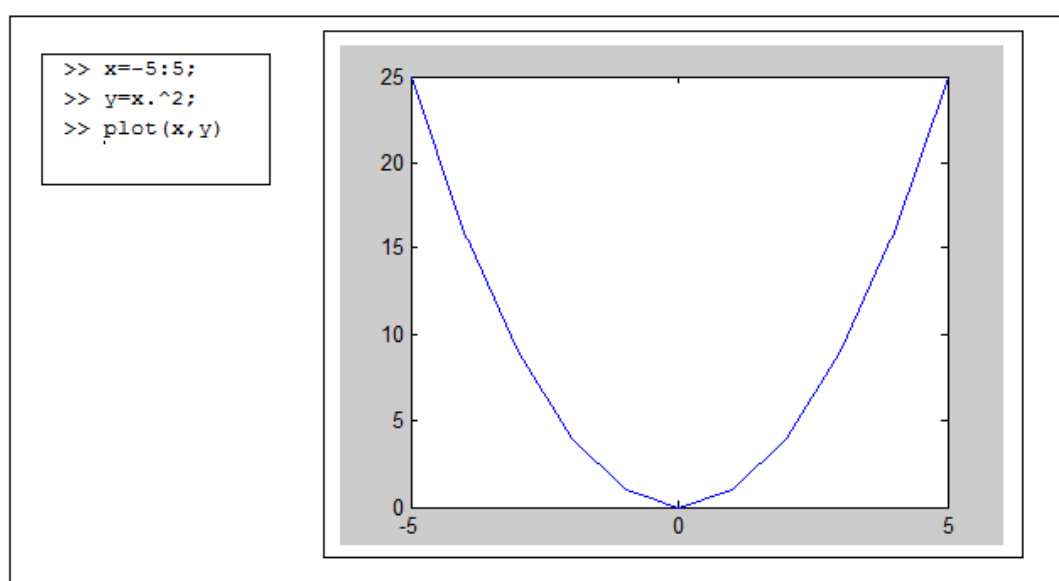


Figura 5 – Representação de um gráfico 2D no MATLAB®

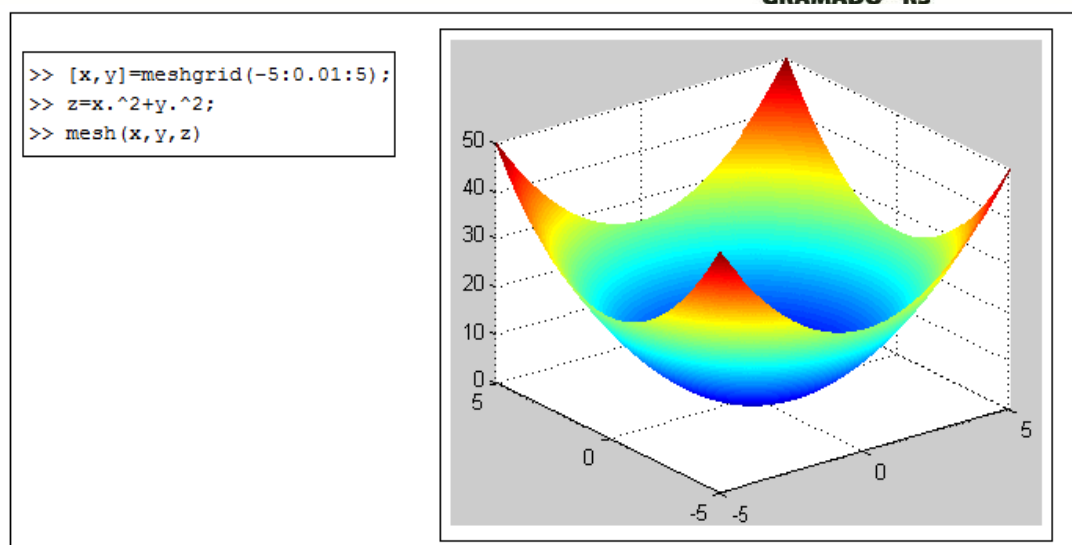


Figura 6 – Representação de um gráfico 3D no MATLAB®

No âmbito do grupo de pesquisa em controle, automação e engenharia do CEFET-MG, durante a realização de projeto de iniciação científica em robótica, podem-se observar algumas falhas no software livre em questão, o Scilab. Falhas de execução de programa são frequentes, onde o mesmo tenha que ser fechado e aberto posteriormente para que volte a funcionar corretamente, o travamento também é constante. As toolboxes robóticas são bem elaboradas, mas não tão completas quanto a de seu concorrente, o Matlab®. No software proprietário as falhas também podem ocorrer, mas com intensidade muito menor do que no software livre.

6. CONSIDERAÇÕES FINAIS

Percebeu-se com esta pesquisa que ambos os softwares são muito importantes para a computação numérica. Há uma grande semelhança entre eles, tendo então lugar para ambos no ramo da engenharia. O Scilab possui uso mais acadêmico, voltado para o aprendizado em cursos de graduação, tendo seu destaque maior por ser gratuito e ter seu código fonte livre, sendo distribuído junto com o mesmo. Assim, todos os alunos podem ter acesso ao programa em seus próprios computadores pessoais sem restrições, enquanto o MATLAB®, software mais abrangente, é mais utilizado em empresas.

Deve-se incentivar o uso do software livre Scilab nas instituições de ensino, onde muitas vezes o MATLAB®, software proprietário, é utilizado em detrimento do primeiro.

Nota-se que o MATLAB® é um software mais bem aperfeiçoado do que o Scilab, este fato pode ser justificando por ser um software pago. Mas as funções básicas principais do Scilab não deixam a desejar quando comparadas com as existentes no MATLAB®.

O MATLAB® e o Scilab são excelentes ferramentas de suporte para as linhas de pesquisa onde o uso de computadores na resolução numérica de problemas é intensivo.

Agradecimentos

Os autores agradecem a FAPEMIG e ao CEFET-MG pelo apoio ao desenvolvimento deste trabalho.

REFERÊNCIAS BIBLIOGRÁFICAS

ALMEIDA, R. Q. Software Livre na Educação. In: Boletim EAD - Unicamp/Centro de Computação/Equipe EAD. http://www.ead.unicamp.br/ead/index_html?foco=Publicacoes, 2002.

DIDIO, L. North American Linux and Windows TCO comparison, Part 1. 2005. The Yankee Group Report, April, 2006. Disponível em: <download.microsoft.com/download/2/8/7/287fda62-1479-808c87333312b93e/Yankee_TCO.pdf>. Acesso em: 29 de maio de 2013.

GNU. Gnu'snot Unix. Disponível em: <<http://web.cecs.pdx.edu/~trent/gnu>>. Acesso em: 28 de maio de 2013.

HEXSEL, R. A. Propostas de ações de Governo para incentivar o uso de software livre. Relatório Técnico do Departamento de Informática da UFPR, n. 004/2002, Curitiba, outubro, 2002. Disponível em: <http://www.inf.ufpr.br/info/techrep/RT_DINF004_2002.pdf>. Acesso em: 31 de maio de 2013.

LAMAS, M. Software livre ao seu alcance. São Paulo: Letras e Letras, 2005.

PERENS, B. The open source definition in open sources: voices from the open source revolution. Illinois: O'Reilly & Associates Inc., 1999.

PIRES, P. S. M., Introdução ao Scilab Versão 3.0. – Apostila - UFRN <http://www.leca.ufrn.br/~pmotta/sciport.pdf>, 2004. Acesso em 30/05/2013.

SALEH, A. M. Adoção de tecnologia: um estudo sobre o uso de software livre nas empresas. 2004. Dissertação (Mestrado em Ciências Econômicas) – Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo. São Paulo: FEA/USP.

SILVEIRA, S. A. da. Software livre: a luta pela liberdade do conhecimento. São Paulo: Fundação Perseu Abramo, 2004.

COMPARATIVE ANALYSIS OF THE USE OF PROPRIETARY SOFTWARE VERSUS FREE SOFTWARE IN UNDERGRADUATE COURSES IN ENGINEERING

Abstract: *The use of numerical computing software is expanding increasingly in engineering. These software's are practical and fast in solving complex problems that are impossible or extremely costly to be solved by humans. The high cost on the acquisition of proprietary software is hindering its use. One solution to this problem is the use of free software. The free software allows users to run, copy, distribute study, change and improve the same, and can adapt it to your needs. Most of the time the free software is free of charge. Already*



proprietary software requires the user to buy a license, with restrictions on its creator in changing its source code. The numerical computation software's most used in engineering courses are MATLAB ® and Scilab, being one paid license and the other one license free, respectively. Both have pretty much the same basic functions, but there are advantages and disadvantages to be discussed between them. This article aims to make a comparison between software's in order to identify the advantages and disadvantages of each one for use in undergraduate courses in engineering.

Key-words: *Free Software, Proprietary Software, MATLAB®, Scilab.*