

**AJES - INSTITUTO SUPERIOR DE EDUCAÇÃO DO VALE DO JURUENA
LICENCIATURA EM MATEMÁTICA**

**O SCILAB COMO FERRAMENTA DE ENSINO: O POTENCIAL DE *SOFTWARES*
LIVRES NA EDUCAÇÃO MATEMÁTICA**

MARCOS EDUARDO CÂNDIDO DOS SANTOS

Juína – MT

2015

**AJES - INSTITUTO SUPERIOR DE EDUCAÇÃO DO VALE DO JURUENA
LICENCIATURA EM MATEMÁTICA**

**O SCILAB COMO FERRAMENTA DE ENSINO: O POTENCIAL DE *SOFTWARES*
LIVRES NA EDUCAÇÃO MATEMÁTICA**

MARCOS EDUARDO CÂNDIDO DOS SANTOS

“Monografia apresentada ao curso de Licenciatura em Matemática, do ISE - Instituto Superior de Educação do Vale do Juruena como exigência parcial para obtenção do título de Licenciado em Matemática”.

Orientador: Prof. Dr. Luciano Endler

**Juína – MT
2015**

**AJES - INSTITUTO SUPERIOR DE EDUCAÇÃO DO VALE DO JURUENA
LICENCIATURA EM MATEMÁTICA**

BANCA EXAMINADORA

Prof.^a. Ma. Aline Fernanda Ventura Sávio Leite

Prof.^o. Me. Fábio Bernardo da Silva

Prof.^o. Dr. Luciano Endler
Orientador

DEDICATÓRIA

À minha família.

AGRADECIMENTOS

Agradeço a todo o corpo docente Instituto Superior de Educação do Vale do Juruena, foi um privilégio ter educadores especializados que, com certeza, contribuíram à formação acadêmica.

Ao professor Luciano Endler, pelas orientações à elaboração deste trabalho e por todo conhecimento gerado nas disciplinas específicas do curso, onde situações complexas são acompanhadas de métodos “muito simples” de resolver.

À minha mãe Cleuzeni Cândida da Silva e meu irmão Maike Henrique Cândido Tenório, por acreditarem que esta formação é essencial para uma vida profissional e por estarem ao meu lado nos momentos de dificuldades.

À professora Lucinda Ap. Américo Honório que foi a maior responsável por ter iniciado esta formação acadêmica.

Enfim, agradeço a todos, que de uma forma ou outra, participaram e apoiaram em algum momento dessa caminhada.

“As tecnologias não são figurantes nos cenários cognitivos. Humanos e tecnologias são protagonistas da ecologia cognitiva.” (Marcelo de Carvalho Borba)

RESUMO

Este trabalho apresenta um estudo teórico sobre a utilização do *software* Scilab no ensino de matemática tendo por finalidade compreender o processo metodológico aplicado ao ensino-aprendizagem com a utilização *softwares* educacionais. Os *softwares* voltados a área da educação matemática proporcionam novas maneiras de expressar e interagir com o conhecimento. Para tanto, busca-se levar em consideração os aspectos técnicos e pedagógicos que são avaliados nos *softwares* educacionais, definindo suas categorias e potencialidades. Este trabalho consiste em uma pesquisa exploratória bibliográfica, visando conhecer o foco das produções acadêmicas voltadas à utilização do Scilab em contexto educacional, os recursos e conteúdos contemplados em sua programação. Além disso, apresenta-se uma breve análise qualitativa dos conteúdos contemplados em sua programação em comparando atividades desenvolvidas no livro didático. Contudo, identifica-se que o Scilab possui um grande número de funções e ferramentas na própria interface voltados a diferentes níveis da educação, porém vem sendo amplamente utilizado no ensino superior com ênfase nas ciências das engenharias. Com comparação teórica com os métodos adotados no livro didático e no quadro branco para construção do gráfico de determinadas funções identifica-se que o Scilab tem potencial para ser aplicado em níveis mais elementares no ensino de matemática. A atividade realizada no Scilab e a interação proporcionada possui grandes proximidades com métodos tradicionalmente utilizados. Isto se dá devido as características algébricas e vetoriais presente em sua programação. Seu *console* lembra uma folha em branco onde o aluno escreve suas ideias, porém apresentando conceitos iniciais de programação. Dessa forma, o Scilab não resolve e fornece respostas prontas ao aluno, mas somente o que se consegue expor para resolver. O seu caráter livre e infinidades de funções coloca-o em comparação com *softwares* de caráter comercial.

Palavras-chave: *Softwares* Educacionais. Educação Matemática. Scilab.

LISTA DE FIGURAS

Figura 1 - Interação aluno-computador (máquina de ensinar).....	17
Figura 2 - Interação aluno-computador (máquina a ser ensinada).....	18
Figura 3 - Interação aprendiz-computador mediada por um <i>software</i> tutorial	25
Figura 4 - Interação mediada por um <i>software</i> tipo Exercícios e Prática	26
Figura 5 - Interação aprendiz-computador na situação de programação	27
Figura 6 - Interface principal do Scilab	39
Figura 7 - Métodos adotados nos livros didáticos para construção gráfica	42
Figura 8 - Métodos utilizados para construção gráfica através do Scilab.....	43
Figura 9 - Finalização do gráfico da Função do 1º Grau	44
Figura 10 - Construção do gráfico da Função do 2º Grau (Livro e Quadro branco) ..	45
Figura 11 - Construção do gráfico da função do 2º grau no Scilab	46
Figura 12 - Reconstruindo o gráfico da função do 2º grau no Scilab	46
Figura 13 - Gráfico inserido através de um <i>software</i> de representação gráfica	48
Figura 14 - Informação de erro do Scilab	49
Figura 15 - Gráfico inserido através no Scilab.....	50
Figura B.1 - Janela de Ajuda e Exemplos	61
Figura B.2 - Janela de Gráficos.....	61
Figura B.3 - Navegador de paleta - Xcos	62
Figura B.4 - Simulador - Xcos	62
Figura B.5 - Janela de Módulos ATOMS.....	63

LISTA DE TABELAS

Tabela 1 - Características da Qualidade de <i>Software</i> segundo a ISO/IEC 9126-1...22	22
Tabela 2 - Aspectos técnicos e pedagógicos observados pelos autores23	23
Tabela 3 - <i>Softwares</i> de folha de cálculo ou algébricos.31	31
Tabela 4 - <i>Softwares</i> de Geometria Dinâmica.....32	32
Tabela 5 - <i>Softwares</i> de representação gráfica.....33	33
Tabela 6 - Lista de funcionalidades do Scilab38	38
Tabela A.1 - Lista de comandos do Scilab59	59

SUMÁRIO

INTRODUÇÃO	10
1.1. Objetivos.....	12
1.2. Organização do trabalho	12
2. A UTILIZAÇÃO DO COMPUTADOR EM SALA DE AULA: UMA BREVE ANÁLISE CONCEITUAL	14
2.1. Substituição, suplementação ou reorganização da prática docente	14
2.2. A educação mediada pelo computador	16
3. SOFTWARES EDUCACIONAIS LIVRES NA EDUCAÇÃO MATEMÁTICA	20
3.1. Avaliação de <i>softwares</i> educacionais em Matemática.....	20
3.2. Classificação de <i>softwares</i> educacionais	24
3.2.1. <i>Softwares</i> tutoriais	24
3.2.2. <i>Softwares</i> de exercícios e práticas	26
3.2.3. <i>Softwares</i> de Programação	27
3.2.4. <i>Softwares</i> de Simulação	28
3.2.5. Jogos Educacionais.....	29
3.3. Classificação de <i>softwares</i> educacionais em matemática	30
3.3.1. <i>Softwares</i> algébricos ou de folha de cálculo.....	30
3.3.2. <i>Softwares</i> de geometria dinâmica.....	31
3.3.3. <i>Softwares</i> de representação gráfica	32
4. METODOLOGIA	34
5. O SOFTWARE SCILAB.....	37
5.1. Estrutura do <i>software</i>	37
5.2. Produções acerca do Scilab	39
5.3. Comparando uma aplicação do Scilab	42
6. CONSIDERAÇÕES FINAIS.....	52
REFERÊNCIAS	55
ANEXO A – Lista de comandos do Scilab.....	59
ANEXO B – Janelas de recursos do Scilab.....	61

INTRODUÇÃO

Ao falar de tecnologia têm-se em mente as grandes invenções criadas pelos homens que provocaram avanços significativos na sociedade. Apesar de ser um termo, teoricamente novo na história da humanidade, a palavra pode ser aplicada a uma margem temporal que acompanha toda trajetória humana ao longo de sua evolução. “Não é por acaso que todas as eras foram, cada uma à sua maneira, “eras tecnológicas”. Assim tivemos a Idade da Pedra, do Bronze... até chegarmos ao momento tecnológico atual [...]” (KENSKI, 2003, p. 02).

Os momentos transitórios proporcionados pela criação de um novo artefato tecnológico sempre causaram receio e desconfiança. O “novo”, tido como algo inexplorado ou que não se tem domínio, causa euforia e preocupação, não necessariamente ao mesmo tempo (BORBA, 2002). Para buscar um exemplo não precisa ir tão longe à história, basta lembrar o momento em que o computador passou a ser uma realidade para muitos no mundo ou, posteriormente, a euforia causada pela criação da internet.

Apesar de já terem sido criados a um certo tempo, ainda se presencia algumas preocupações geradas por esta tecnologia, algumas ligadas a desigualdade tecnológica, inclusão digital e outras à educação proporcionada por tal aparato. Assim, alguns questionamentos surgiram ou ganharam novos ares tais como: qual será o papel do professor em uma educação mediada por tal tecnologia? E do aluno?

É sabido que a ampliação das possibilidades de comunicação e de informação alterou a forma de viver, adaptando e criando novas situações. Dessa forma, Borba; Scucuglia e Gadanidis (2014) afirmam que o ser humano molda uma determinada tecnologia e, ao mesmo tempo, é moldado por ela.

Com o atual momento propiciado pelas TICs (Tecnologias de Informação e Comunicação) e pelas NTICs (Novas Tecnologias de Informação e Comunicação) vislumbra-se a rapidez e a capacidade de armazenamento e processamento de informações geradas como apenas uma ferramenta. Logo, os novos desafios estão ligados em como tirar proveito destes benefícios nos setores da educação.

Este trabalho traz, em especial, a utilização de *softwares* educacionais no ensino de matemática. A produção deste tipo de ambiente, destinadas a várias áreas do conhecimento, vem aumentando nos últimos anos. Para Borba; Scucuglia e Gadanidis (2014) os *softwares* trazem, quando utilizados corretamente, novas maneiras de expressar e visualizar o conhecimento matemático, qualitativamente diferentes comparando-se com o lápis e papel.

Segundo o artigo quinto, inciso sexto da Resolução Conselho Nacional de Educação, CNE/MEC nº 2, de 01 de julho de 2015 a formação de professores deve estar ligada, entre outros aspectos, “ao uso competente das Tecnologias de Informação e Comunicação (TIC) para o aprimoramento da prática pedagógica e a ampliação da formação cultural dos (das) professores (as) e estudantes”(BRASIL, 2015). Mas o que seria uso competente de tecnologia? Nos Parâmetros Curriculares Nacionais (PCNs) consta que:

Quanto aos *softwares* educacionais é fundamental que o professor aprenda a escolhê-los em função dos objetivos que pretende atingir e de sua própria concepção de conhecimento e de aprendizagem, distinguindo os que se prestam mais a um trabalho dirigido para testar conhecimentos dos que procuram levar o aluno a interagir com o programa de forma a construir conhecimento (BRASIL, 1998, p. 32).

Logo, o uso competente de tecnologias, neste caso *softwares*, está intimamente ligado ao conhecimento que o docente tem sobre esta ferramenta, não somente o saber manusear, mas também saber avaliar e identificar as potencialidades de cada *software*, bem como o que se espera obter do aluno com a utilização desta tecnologia.

Dentre os vários *softwares* disponíveis escolheu-se o *software* Scilab que constitui o *software* algébrico ou “folha de cálculo”. É distribuído gratuitamente aos usuários e está em constante desenvolvimento. Está entre os *softwares* mais utilizados nas ciências das engenharias, comparado até com *softwares* comerciais hegemônicos utilizados no mercado profissional.

Seguindo as recomendações propostas nos PCNs e a exigência da resolução CNE/MEC perante o uso de tecnologias, faz-se necessário que o uso das tecnologias seja analisado e adaptado de acordo com os conteúdos e objetivos que se deseja alcançar. Para tanto, este trabalho delimita-se a exploração do *software*

Scilab em contexto educacional, identificando produções acadêmicas, conteúdos contemplados e métodos de utilização.

1.1. Objetivos

O presente trabalho tem como objetivo geral compreender o processo metodológico aplicado ao ensino-aprendizagem com a utilização *softwares* matemáticos.

Para isto, adotaram-se os seguintes objetivos específicos:

- Identificar tipos de *softwares* educativos e quais são suas especificidades;
- Mostrar como vem sendo utilizado o Scilab na educação, visando dar visibilidade ao *software*;
- Apresentar os conteúdos e recursos contemplados pelo Scilab;
- Propor formas de utilização do Scilab em sala de aula, comparando-as com o livro didático e com outros *softwares*;

1.2. Organização do trabalho

O trabalho está organizado em 6 capítulos. No Capítulo 2 descreve-se a utilização do computador em sala de aula e as implicações geradas a prática docente, bem como a diferença proporcionada ao computador como “máquina de ensinar” e “máquina a ser ensinada”. No Capítulo 3 aborda-se os aspectos avaliados em um *software* educacional, levando em consideração a parte técnica e pedagógica. Apresenta-se como são classificados

O Capítulo 4 traz, primeiramente, uma análise da estrutura do *software* Scilab, conteúdos e recursos disponíveis. Será exposto também a revisão de literatura sobre a utilização do Scilab em contexto educacional. Apresenta-se uma comparação do Scilab com aplicações, usualmente, realizados livros didáticos no Ensino Fundamental e em outros *softwares*.

No Capítulo 5 mostra-se o percurso metodológico da pesquisa, ligando a abordagem adotada aos resultados obtidos. Apresenta-se também as técnicas utilizadas para dados coletados. E, finalmente, no Capítulo 6 apresentam-se as conclusões e perspectivas para o desenvolvimento de trabalhos futuros.

2. A UTILIZAÇÃO DO COMPUTADOR EM SALA DE AULA: UMA BREVE ANÁLISE CONCEITUAL

Neste capítulo encontram-se algumas ideias voltadas para a utilização do computador como proposta para o ensino-aprendizagem. Para tanto, a seguir apresenta-se uma breve análise das ideias referidas ao computador em ambiente escolar e a prática pedagógica que permeia a sua utilização.

2.1. Substituição, suplementação ou reorganização da prática docente

A utilização do computador em sala de aula ou, ao menos, incorporada a prática docente não é algo novo a se pensar. Apesar de ter um início destinado apenas a algumas universidades americanas a euforia relacionada ao computador na educação começou a mostrar lados positivos e negativos a medida que começou a ganhar espaço. Dessa forma, as implicações que o computador poderia causar no trabalho docente e na sociedade em geral começaram a ser debatidas.

Em um trabalho publicado em 1981, “*The Psychological Consequences of Computerization*”¹, Tikhomirov confrontou ideias emergentes sobre a implicação do computador na sociedade, formulando linhas de raciocínios que descreviam o pensamento e, naquela época, apresentava uma visão mais futurista sobre o caminho que deveria ser seguido quanto a utilização desta tecnologia. A primeira teoria a ser confrontada é a da “substituição”, onde o computador seria responsável por substituir o ser humano nas atividades intelectuais em várias áreas do conhecimento (TIKHOMIROV, 1981).

Tal raciocínio pode ser evidenciado ao analisar o início da inserção do computador em sala de aula, onde é possível perceber que a sua utilização era uma simulação informatizada da prática docente da época (BORBA, 1999; VALENTE, 1999). Isso ocorreu devido os primeiros *softwares* destinados à educação possibilitarem armazenamento de informações e liberá-las em módulos para o aluno. A progressão aos módulos subsequentes era dada por avaliação final e o resultado era liberado rapidamente pelo computador. Este tipo de programa, conhecido como ensino

¹ As consequências psicológicas da informatização.

programado, basicamente, substituía a atividade docente em sala de aula e, portanto, não ocasionou uma mudança significativa na prática pedagógica (VALENTE, 1999).

O *software* foi desenvolvido, inicialmente, nos Estados Unidos pelo psicólogo americano Burrhus Frederic Skinner no início da década 1950, baseado em uma linguagem do tipo CAI (*Computer Aided Instruction*²). Valente (1993) caracterizou-o como “máquina de ensinar” ligada perspectiva educacional instrucionista e apresentou, assim como Borba (1999), que o maior receio em utilizar esta tecnologia era de acabar substituindo o trabalho docente e extinguir a profissão. Além disso, o medo que o papel da educação fosse desumanizado se tornou uma realidade para a época.

Confrontando esta teoria Tikhomirov (1981) afirma que o computador não atua na seleção dos problemas do ser humano e sim na resolução dos mesmos. Dessa forma, todos os critérios de seleção de conteúdos adotados pelo homem são ignorados, não havendo um discernimento entre conceitos, revisão de conceitos e critérios de avaliação. Todo conhecimento armazenado nos programas do tipo CAI são baseados em escolhas do próprio ser humano, que julga quais as potencialidades e finalidades que se pretendem alcançar nos alunos. Sendo assim, tal teoria não pode ser sustentada.

A segunda teoria a ser confrontada diz respeito à “suplementação”, onde o computador complementar os processos humanos ampliando o armazenamento e processamento de informação (TIKHOMIROV, 1981). A atividade intelectual seria então fragmentada entre o ser humano e o computador, a soma dos dois constitui o conhecimento em sua totalidade. Nesta perspectiva, a utilização do computador poderia causar uma mudança significativa na prática docente, pois era considerado aliado à educação. Porém, a ideia de que o conhecimento seria separado em homem e máquina não agradou a sociedade acadêmica.

Ao terem a ilusão de reduzir o pensamento em a “pequenas caixas” não consideram que o processo de busca de um problema e de busca de solução para este problema [...] não pode ser decomposto e sim deve ser entendido de forma global. Mais ainda, este modelo de pensamento ignora que há valores que perpassam tanto a eleição de um dado problema como suas possíveis soluções (BORBA, 1999, p. 287).

² Instrução assistida por computador.

Logo, a teoria da suplementação sai de cena e entra a terceira teoria a ser confrontada, a “reorganização”, onde o computador é visto como um agente regulador e que molda as atividades desenvolvidas pelo homem (TIKHOMIROV, 1981). Para tal teoria, o pensamento humano pode ser organizado de novas maneiras, possibilitando abordagens inovadoras e desenvolvimento de novos saberes. Tudo dependerá de como o computador será manuseado. Nesta perspectiva o computador pode constituir uma nova maneira de expressar e representar o conhecimento, mostrando interdependência entre os mesmos (BORBA, 1993 apud BORBA; SCUCUGLIA e GADANIDIS 2014).

A partir dessa visão, autores como Valente (1999), Borba; Scucuglia e Gadanidis (2014), Kenski (2012) e Valente (1997b) deslumbram a real possibilidade de utilização do computador em sala de aula. Evidenciam o quão significativo um conhecimento gerado pode se tornar com sua utilização e como novos problemas podem se tornar pedagógicos ao serem comparados com outras práticas metodológicas.

2.2. A educação mediada pelo computador

É imprescindível que professores de todo o corpo docente estejam alerta às exigências que a sociedade atual faz, aliando-se ao processo evolutivo tecnológico que se modifica a cada momento, a realidade sociocultural em que as pessoas e em particular, os alunos, estão inseridos (KENSKI, 2003).

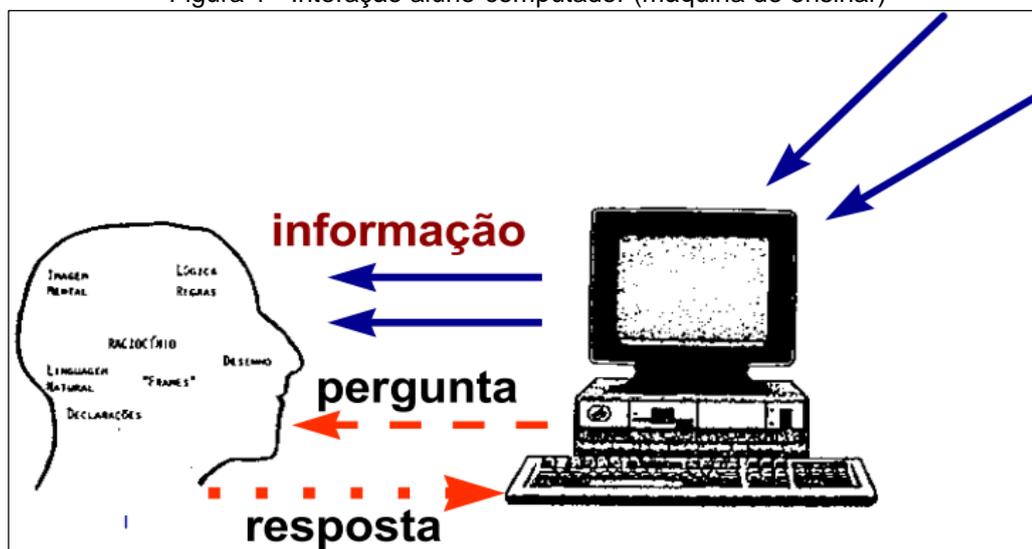
Segundo Valente (2005) e Borba; Scucuglia e Gadanidis (2014) a compreensão do docente sobre o que é o conhecimento implicará na visão que possui sobre as tecnologias voltadas à educação. Segundo os autores as diferenças entre conhecimento e informação, tratadas nas diversas teorias do conhecimento, devem ser levadas em consideração ao analisar a aplicação de uma tecnologia em contexto educacional. Ao diferenciar conhecimento de informação Valente afirma:

A informação será tratada [...] como os fatos, os dados que encontramos nas publicações, na internet ou mesmo aquilo que as pessoas trocam entre si. Assim, passamos e trocamos informação. O conhecimento é o que cada indivíduo constrói como produto do processamento, da interpretação, da compreensão da informação. É o significado que atribuímos e representamos em nossa mente sobre a nossa realidade (VALENTE, 2005, p. 4).

Ao analisar o computador como tecnologia educacional o conhecimento, conforme a sua utilização, pode se aproximar de duas das renomadas teorias de ensino-aprendizagem, apresentadas em extremidade opostas tem-se a instrucionista e a construtivista (VALENTE, 1997a; BORBA; SCUCUGLIA; GADANIDIS, 2014).

Se o ensino se basear apenas na transmissão informação e a interação aluno-computador se resumir apenas a este vai e vem de informação, o computador assume papel de máquina de ensinar (VALENTE, 1993). Isto se aproxima do ensino instrucionista, onde o computador será responsável por instruir o aluno através de informações pré-selecionadas, conforme apresentado na Figura 1.

Figura 1 - Interação aluno-computador (máquina de ensinar)



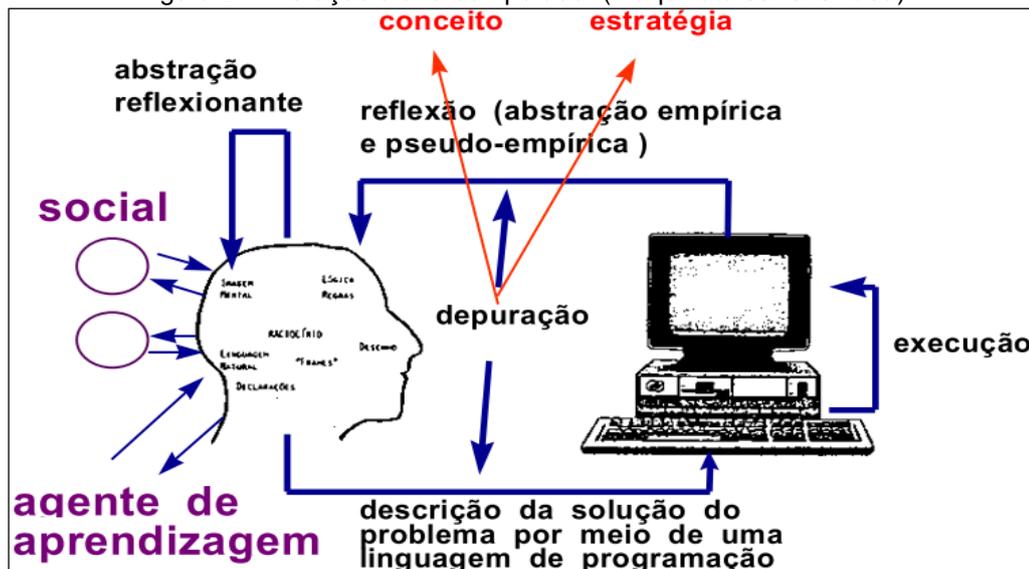
Fonte: Extraído de VALENTE (1997a, p. 1).

Nesta visão do ensino, o papel do professor é delimitado à seleção de conteúdos e problemas que serão inseridos no computador (PENTEADO et al., 2000; KENSKI, 2003). A transmissão de informação e processo avaliativo será realizada pelo computador remetendo-se a primeira teoria de Tikhomirov supracitada. O conhecimento não sofre nenhum aditivo positivo ou, até mesmo, negativo por parte do aluno e o saber passa se estagnar.

Se o aluno possuir um posicionamento crítico em relação à tecnologia em questão e utilizá-la como uma nova maneira de representar suas ideias, então o caminho percorrido pela informação será realizado ao contrário. A informação partirá inicialmente do aluno, baseada em um conhecimento previamente definido por ele, para o computador (VALENTE, 1993; 1997b).

Dessa forma, autores como Valente (1993), Borba e Penteado (2002), Kenski (2012) e Borba, Scucuglia e Gadanidis (2014), apontam que conhecimento aproxima-se do construtivismo³. O aluno constrói o conhecimento e atua ativamente neste processo, depurando e aprimorando conceitos, formulando estratégias e buscando representar suas ideias em uma linguagem compreensível. Valente (1993) chamou esta interação de máquina a ser ensinada⁴ como representado na Figura 2.

Figura 2 - Interação aluno-computador (máquina a ser ensinada)



Fonte: Extraído de VALENTE (1997a, p. 1).

A imagem citada basicamente representa um ciclo construtivo que o aluno deve percorrer ao construir o seu conhecimento. Neste caso o posicionamento do professor deve propiciar estratégias que vise mantê-lo neste ciclo. Segundo Valente (1993), Penteado et al. (2000) e Ribeiro e Ponte (2000) o professor deverá, nesta visão, analisar se a tecnologia está contribuindo no processo de aprendizagem e propiciar um ambiente que colabore para o processo, como já havia sido dito por Valente:

³ A Teoria Construtivista, surgiu no século XX, a partir das experiências de Jean Piaget (1896-1980). Em síntese, tal teoria parte da ideia de que o conhecimento não é algo pronto e acabado, muito menos é dado ou transmitido de um ser para outros. Nesta perspectiva o conhecimento é fruto da interação do indivíduo com o meio físico e social. Ou seja, o indivíduo procura encontrar respostas a partir de seus próprios conhecimentos e de pela interação com a realidade e com seus semelhantes. (COLL, 1997). Baseando-se na ideias de Jean Piaget e voltando-se a utilização do computador com ferramenta de ensino Papert (1980 apud VALENTE, 1997a), idealizador a linguagem de programação Logo, denominou abordagem construtivista aquela onde o aluno constrói seu próprio conhecimento por intermédio do computador. Neste caso o computador é algo agregado ao meio e a realidade do indivíduo.

⁴ Valente (1999) utiliza a mesma notação e imagem para demonstrar a interação com os softwares de programação que a serão apresentados no decorrer do trabalho.

A mudança da função do computador como meio educacional acontece juntamente com um questionamento da função da escola e do papel do professor. A verdadeira função do aparato educacional não deve ser a de ensinar, mas sim a de criar condições de aprendizagem. Isto significa que o professor deve deixar de ser o repassador do conhecimento [...] e passar a ser o criador de ambientes de aprendizagem e o facilitador do processo de desenvolvimento intelectual do aluno (VALENTE, 1993, p. 6).

Para que possa ter um posicionamento crítico relacionado a tecnologia o professor deve: (i) conhecer o processo de construção de conhecimento; (ii) e deve conhecer diferente modalidade para a utilização da informática na educação e suas potencialidades. No capítulo seguinte apresenta-se uma das modalidades para o uso da informática na educação conhecida como *softwares* educacionais⁵.

⁵ A historicidade do computador, bem como a abordagem e propostas educacionais adotadas em diversos países pode ser encontrada nos trabalhos desenvolvidos por Valente e Almeida (1997), Valente (1999) e Almeida (2008). Referente ao Brasil, os autores trazem onde o país se sobressaiu em relação aos demais e apresentam os programas desenvolvidos para encaminhar a utilização do computador nas escolas.

3. SOFTWARES EDUCACIONAIS LIVRES NA EDUCAÇÃO MATEMÁTICA

Neste capítulo apresenta-se os aspectos levados em consideração ao avaliar um *software* educacional. Será possível compreender como são categorizados mediante aos conteúdos e interações geradas com o usuário.

3.1. Avaliação de *softwares* educacionais em Matemática

Software, de forma geral, é o que permite a comunicação entre o homem e máquina. Para caracterizar um *software* educacional este deve estar inserido num contexto de ensino-aprendizagem, apresentar fins educacionais e estar ligado a finalidades que vão desde a aquisição de conceitos, passando pelo desenvolvimento de habilidades e até resolução de problemas (PENTEADO; BORBA; GRACIAS, 1998; VALENTE, 1999)

De acordo Silveira (2003) o *software* é de caráter livre se possui código fonte aberto, permitindo ao usuário executar o programa para qualquer propósito; de estudar como o programa funciona e adaptá-lo as suas necessidades; de redistribuir cópias; e de aperfeiçoar, liberando e incorporando os seus aperfeiçoamentos, a fim de beneficiar os usuários.

Ao afirmar que “muitas vezes um problema que poderia ser didático com uma tecnologia não é com outra” (BORBA, 2014, p. 24-25), tem-se em mente que as ferramentas utilizadas a favor da educação possuem suas limitações. Isto consiste em uma busca por metodologias e recursos que visam diminuir este percalço.

Atualmente os *softwares* vêm ganhando espaço dentre as metodologias e recursos disponíveis ao trabalho docente (GOMES et al., 2002). A grande variedade e as diferentes abordagens que vêm apresentando faz destes um dos mais amplos recursos que pode salientar a busca por recursos agregados a prática pedagógica. A questão está em reconhecê-los como um potencial à educação, já que:

Para cada *software* há uma descrição, uma análise e uma avaliação, a fim de que estudantes e professores possam obter informações que lhes permitam optar por um outro *software*, de modo a atender melhor às suas necessidades (PENTEADO, 1998, p. 83).

Dessa forma, o docente será responsável por avaliar tanto a situação de aprendizagem em que o aluno se encontra como os *softwares* que deseja utilizar como ferramenta de ensino-aprendizagem. No entanto, *softwares* podem pertencer à mesma categoria, mas não contemplar os mesmos objetivos quando aplicados em um determinado contexto (GOMES et al., 2002; MARTINS, 2002).

Mascarenhas (2009) e Oliveira (2008) apontam que o estudo entorno dos *softwares* deve englobar tanto a parte técnica quanto a parte pedagógica. Em pesquisas desenvolvidas, o simples saber manusear o *software* não possibilitou uma aprendizagem ligada a esta metodologia. Isto induz que o ensino mediado por esta tecnologia pode dar certo ou não, dependerá de como será a utilização.

Compreender a singularidade de cada *software* e suas potencialidades pedagógicas é o essencial para utilizá-lo como ferramenta de ensino-aprendizagem. Tal conhecimento deve ser estruturado ao longo da vida profissional, começando pela formação acadêmica, pois pode envolver vários aspectos a serem analisados que, talvez, não serão possíveis estruturar em um breve planejamento de aula. (BORBA; SCUCUGLIA; GADANIDIS, 2014).

Para a utilização de *softwares* como ferramentas da prática pedagógica o docente deve saber avaliá-lo e identificar suas potencialidades. Na literatura há vários autores que levantam características que devem ser verificadas em um determinado *software*.

Segundo Gladcheff e Zuffi (2001) e Gomes et al. (2002) os *softwares* educacionais eram avaliados, tradicionalmente, a partir da estrutura de sua construção e estavam relativos à: qualidade da interface e coerência de apresentação dos conceitos e aos aspectos ergonômicos gerais. Esta avaliação é chamada de “avaliação de processo e produto”⁶ onde as características avaliadas são: funcionalidade, usabilidade, confiabilidade, eficiência, manutenibilidade e

⁶ “Avaliação do Processo” consiste no exame dos procedimentos operacionais e gerenciais, métodos e técnicas utilizados nas fases de desenvolvimento de um produto de software, com o objetivo de identificar práticas que possam provocar problemas na qualidade do produto e de estabelecer novas práticas que evitem estes problemas. “Avaliação do Produto” consiste no exame de um produto final resultante de um Processo de Desenvolvimento de Software, ou de produtos resultantes de atividades de fases intermediárias deste processo (TSUKUMO et al., 1997; GLADCHEFF, 2001).

portabilidade (ISO/IEC⁷ 9126-1,1997 apud Tsukumo, 1997). Na Tabela 1 são apresentadas suas definições.

Tabela 1 - Características da Qualidade de *Software* segundo a ISO/IEC 9126-1.

CARACTERÍSTICAS	DESCRIÇÃO
Funcionalidade	Evidencia que o conjunto de funções atendem às necessidades explícitas e implícitas para a finalidade a que se destina o produto.
Confiabilidade	Evidencia que o desempenho se mantém ao longo do tempo e em condições estabelecidas.
Usabilidade	Evidencia a facilidade para a utilização do <i>software</i> .
Eficiência	Evidencia que os recursos e os tempos envolvidos são compatíveis com o nível de desempenho requerido para o produto.
Manutenibilidade	Evidencia que há facilidade para correções, atualizações e alterações.
Portabilidade	Evidencia que é possível utilizar o produto em diversas plataformas com pequeno esforço de adaptação.

Fonte: Extraído de Tsukumo et al. (1997, p.13).

Para Gladcheff e Zuffi (2001) e Ramos e Mendonça (1991) esses aspectos técnicos devem ser levados em consideração, mas não tidos como as únicas prioridades. O tratamento pedagógico que o *software* recebe deve levar em conta a situação e o público que se deseja trabalhar, neste caso o aprendiz.

Na busca por referencial que salientasse as características pedagógicas que devem ser avaliadas em um *software* na área da educação matemática os trabalhos de Gladcheff (2001), Gomes et al. (2002), Ramos e Mendonça (1991), Martins (2002) e Rocha e Campos (1993) foram os que trataram de forma abrangente, não se remetendo a um *software* em específico. Esta avaliação da ferramenta educacional tem por objetivo segundo Gomes et al. (2002):

(a) auxiliar educadores a construir representações mais precisas a respeito da adequação dos materiais disponíveis à sua prática docente;(b) o desenvolvimento de *softwares* educativos, permitindo focalizar o conteúdo a ser mediatizado, além do exame de sua usabilidade; (c) [...] a simplificação do processo de depuração das interfaces, antes de sua implementação (GOMES, 2002, p. 3).

⁷ ISO: *International Organization for Standardization* (Organização Internacional para Padronização); IEC: *International Electrotechnical Commission* (Comissão Eletrotécnica Internacional).

Para integrar os *softwares* educacionais à prática pedagógica a avaliação de sua qualidade deve ir além de questões padronizadas e meramente técnicas, acrescentando e englobando aspectos ideológicos e psicológicos. Na Tabela 2 são apresentados os aspectos que devem ser observados em um *software* educacional segundo os autores.

Tabela 2 - Aspectos técnicos e pedagógicos observados pelos autores

ASPECTOS TÉCNICOS	Documentação	Apresentar instruções (manual) de uso, de instalação e desinstalação - descrição de todas as funções.
	Especificidade	Requisitos mínimos necessários (<i>hardware e software</i>) para instalação - Integridade compatível com a descrição do <i>software</i> - Suporte técnico (fornecedor);
ASPECTOS PEDAGÓGICOS	Objetivos	Especificar os objetivos que pretende alcançar - possuir ao menos um dos itens: Projeto ou Manual Pedagógico/Plano de Ensino/Proposta Educacional - atender a realidade do aluno – valorizar: diferentes formas de resolução, troca de experiência, atividade cooperativa e grupal – integridade da linguagem Matemática.
	Usabilidade	Interface adequada a faixa etária – representação de fácil compreensão e utilização – orientação claras e precisas – se apresenta oscilações na clareza das representações – depurações e especificações de erros - se a interface possui “sistema de ajuda”.
	Conceitos	Apresentar os conceitos matemáticos que pretende trabalhar – se permite que um conteúdo seja desconsiderado pelo professor caso não se deseje trabalhá-lo – se permite refletir sobre a possibilidade dos conceitos matemáticos com ou sem o <i>software</i> – se a abordagem é compatível com as concepções do professor.
	Praticidade	Verificar se o produto possui uma versão para ser utilizado em rede; verificar se seu preço é compatível com o orçamento da escola - verificar se o produtor recolhe sugestões e/ou reclamações tanto por parte do professor quanto do aluno.

Fonte: Elaborado pelo autor.

Avaliar a qualidade de *softwares* educacionais é um passo necessário para o controle e a maior garantia para se alcançar um ensino qualificado. Com isso, novas estratégias metodológicas exigem pesquisas voltadas a busca de novas

experiências didáticas e da qualificação dos profissionais que atuam na educação. A utilização sem conhecimento específico e metodológico pode ocasionar o que Borba (1999) chamou de “domesticação da tecnologia”, onde a tecnologia é utilizada para aplicar práticas anteriores.

3.2. Classificação de *softwares* educacionais

Ao realizar o processo avaliativo é possível identificar singularidades em cada *software*, formatos ergonômicos, conteúdos e aplicações. Com isso, é possível organizá-los em grupos e subgrupos, conforme as semelhanças. Entender como os *softwares* são classificados também ajuda o docente na hora de escolher qual é o mais apropriado para uma determinada situação. Se tratando de computador em sala de aula Valente afirma que:

A análise dos diferentes usos do computador na educação nos permite concluir dois resultados importantes. Primeiro, que o computador pode tanto passar informação ao aprendiz, quanto auxiliar o processo de construção do conhecimento e de compreensão do que fazemos. Segundo, que implantar computadores nas escolas sem o devido preparo de professores e da comunidade escolar, não trará os benefícios que esperamos (VALENTE, 1999, p. 83).

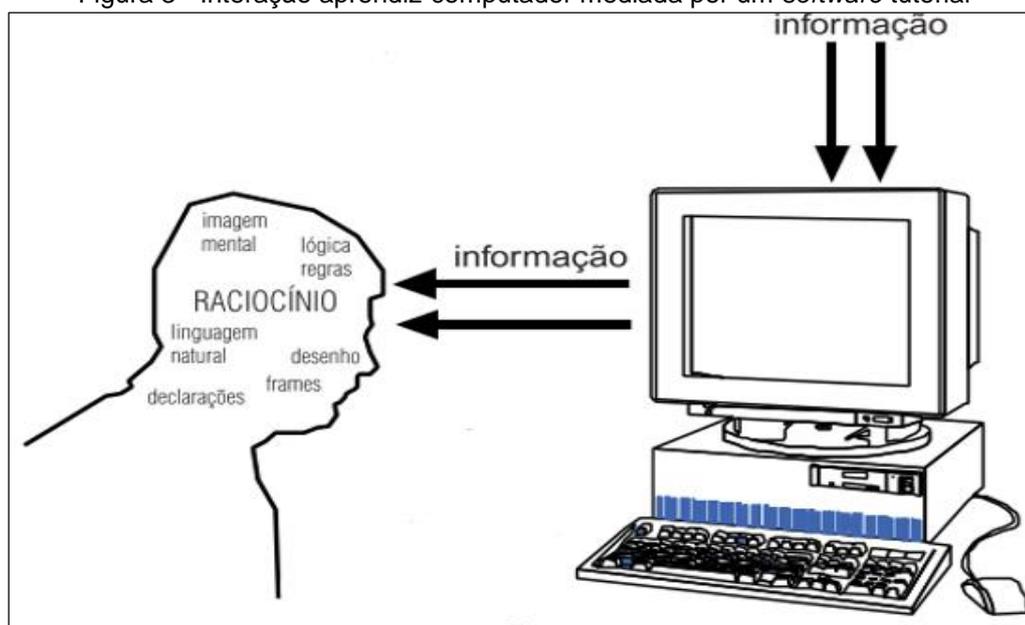
As estratégias utilizadas na classificação dos tipos de *softwares* educacionais são inúmeras, mas acabam se remetendo ao processo avaliativo citado anteriormente, levando em consideração: a característica, o funcionamento, a abordagem pedagógica, os níveis de aprendizagem e os objetivos pedagógicos (VALENTE, 1999; OLIVEIRA; COSTA; MOREIRA, 2001; JUCÁ, 2011). Algumas classificações apresentadas em comum pelos autores indicam *softwares* dos tipos: Tutoriais, Exercícios e Prática, Programação, Simulação e Jogos (existem outras classificações).

3.2.1. *Softwares* tutoriais

Tutoriais são *softwares* no qual a informação é organizada de acordo com uma sequência pedagógica particular. O aprendiz/usuário pode seguir uma sequência determinada ou, em alguns casos, escolher uma sequência de seu interesse. “A interação entre o aprendiz e o computador consiste na leitura da tela ou

escuta da informação fornecida, avanço pelo material, apertando a tecla *ENTER* ou usando o *mouse* para escolher a informação” (VIEIRA, 1999, p.5). Neste caso, Valente (1999) afirma que o computador assume o papel de “máquina de ensinar” e está enraizado no ensino programado. Na Figura 3 é apresentada a interação fornecida por um *software* com estas características.

Figura 3 - Interação aprendiz-computador mediada por um *software* tutorial



Fonte: Adaptado de VALENTE, 1999, p.72.

Autores como Vieira (1999) e Oliveira; Costa e Moreira (2001) afirmam este tipo de *software* também é conhecido como “*softwares* de informação” por somente transmitir informação ou ensinar um determinado procedimento. Apresenta-se como versão animada de um livro, encaixando efeito audiovisuais e caracterizando um professor eletrônico. A crítica está na posição que o professor assume:

Esse programa só permite ao "agente de aprendizagem" verificar o produto final e não os processos utilizados para alcançá-lo. A sua limitação se encontra justamente em não possibilitar a verificação se a informação processada passou a ser conhecimento agregado aos esquemas mentais (VALENTE, 1999, p. 73).

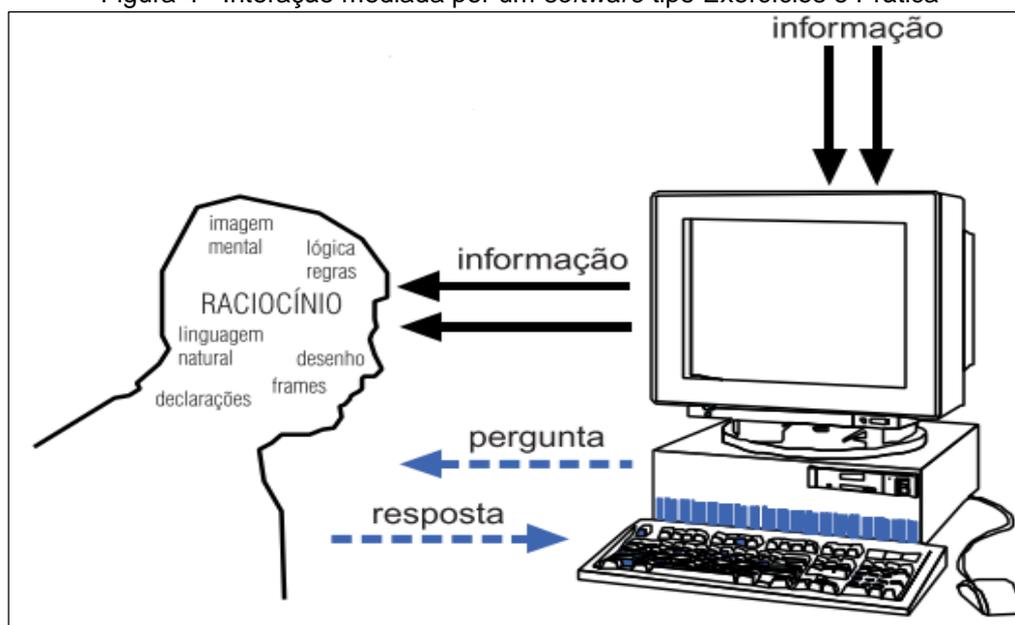
Apesar da possibilidade de repetir a atividade inúmeras vezes no *software* favorecer o aprendiz/usuário que apresenta maior dificuldade Jucá (2011) afirma que os conteúdos, neste tipo de *software*, são limitados pela equipe de desenvolvedores em processo de sua construção.

3.2.2. Softwares de exercícios e práticas

Em geral os *softwares* de exercícios e práticas apresentam exercícios para serem resolvidos, onde as perguntas e respostas são normalmente utilizadas para revisar material já estudado. Segundo Valente (1993) estes *softwares* caracterizam por estimular a memorização e repetição de um determinado conteúdo, tornando-se algo mecânico. Por conter módulos e exercícios previamente selecionados estes softwares podem assemelhar-se ao ensino programado.

As vantagens apresentadas em Valente (1993) e Oliveira; Costa e Moreira (2001) mostram que este tipo de *software* pode proporcionar exercícios em diferentes graus de conhecimento, coletar respostas e emitir resultados avaliativos, diferenciando-se do *software* tipo tutorial. A Figura 4 mostra a relação entre aluno-computador em detrimento do *software* tipo Exercícios e Prática.

Figura 4 - Interação mediada por um *software* tipo Exercícios e Prática



Fonte: Extraído de VALENTE (1999, p.72).

Após o resultado ser emitido, o professor terá em mente os conceitos “absorvidos” pelo aluno. Entretanto, avaliar é um processo complexo e os dados fornecidos pelo *software* pode não ser o suficiente. Saber que o aluno acertou ou errou torna-se facilmente perceptível, mas compreender o porquê de tal resultado exige um posicionamento do professor perante esta interação.

3.2.3. Softwares de Programação

Softwares de Programação são que permitem criar programas através de uma linguagem de programação, ou seja, baseiam-se em um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador permitindo ao usuário comunicar-se como o computador através de instruções. Dentre as linguagens programação amplamente utilizadas estão: Java, PHP, Pascal, C+, C++.

Segundo Valente (1999) estes *softwares* apresentam-se como uma ferramenta para resolver problemas, onde o aprendiz utiliza-o para representar suas ideias. O processo consiste em desenvolver um programa no próprio *software* para resolver um determinado problema utilizando conceitos, estratégias e informações processadas e descritos através do computador.

Dessa forma, o aluno precisará descrever a situação em uma linguagem compreensível tanto para ele como para o computador (programação); executar esta linguagem visando obter a solução; refletir sobre a linguagem utilizada e a solução obtida; depurar as informações geradas e, enfim, descreve-las no *software* (VALENTE, 1997b; 1999; OLIVEIRA; COSTA; MOREIRA, 2001). A Figura 5 mostra o processo de interação mediado por um *software* do tipo programação.



Fonte: Extraído de Valente (1999, p.75).

Este ciclo envolvendo descrição-execução-reflexão-depuração-descrição direciona o aluno à interação direta, no sentido do aluno para o computador, caracterizando-o com uma “máquina a ser ensinada”(VALENTE, 1999). Este ciclo poderá auxiliar o aluno encontrar seus erros e o professor a compreender os conceitos expressos por meio do software. O posicionamento do professor com a “máquina a ser ensinada” já foi apresentado na secção 2.2.

3.2.4. Softwares de Simulação

Consiste em um tipo específico de *software* que representa determinado fenômenos por meio do computador (VALENTE, 1999). Esta simulação pode se apresentar de duas formas: simulação aberta, caracterizadas por situações parcialmente ou totalmente desenvolvidas pelos alunos; e simulação fechada, restringida a um fenômeno previamente implementado no computador.

Quando a simulação é conduzida inteiramente pelo aluno, auxiliado pelo professor ou não, Valente (1999) afirma que esse processo pode constituir duas partes que favorecem a construção do conhecimento: modelagem e programação. Primeiramente, há a criação de um modelo onde o aluno participa inteiramente da escolha do fenômeno real a ser estudado e da solução. Depois o aluno busca implementar recursos computacionais para simular e verificar a proximidade com o contexto pesquisado.

Oliveira; Costa e Moreira (2001) acreditam que este é o extremo da solução aberta, o aprendiz participará do processo de criação e representação (através de sistemas computacionais) de um modelo real. Mas este não é a única forma que se apresenta a simulação aberta. Mesmo não sendo de total criação do aprendiz e levar algo previamente definido (criação da simulação por outros) o fato de apresentar aberturas para o aluno modificar a estrutura do fenômeno e da programação consiste no conceito de simulação aberta.

As atividades desenvolvidas neste tipo de *software* podem caracterizar aspectos do *software* do tipo programação, onde o ciclo de interação, descrição-execução-reflexão-depuração-descrição, deve ser mantido. Nesse caso, o professor

tem o papel de auxiliar o aprendiz, criando condições transitem entre o fenômeno real e a simulação.

No contraponto, a “simulação fechada” consiste de um fenômeno é previamente definido no computador. As informações, parâmetros e programação são inalteradas. A passividade do aprendiz perante simulação se assemelha a um *software* do tipo tutorial, limitando suas ações e criatividade (VALENTE, 1999; OLIVEIRA; COSTA; MOREIRA, 2001).

Os autores podem tratar “simulação fechada”, “simulação aberta”, modelagem e programação como algo totalmente diferente devido as suas características e potencialidades. Entretanto, está se tornando cada vez mais comum os *softwares* apresentarem uma mescla em vários tipos de *softwares*, visando compartilhar suas potencialidades e minimizar suas desvantagens.

3.2.5. Jogos Educacionais

Jogos Educacionais são elaborados especificamente para ensinar as pessoas sobre determinado assunto, baseando-se em uma linguagem de simples compreensão. Possuem como ponto principal o lúdico e o divertimento, levando o aprendiz a um espírito de competição com a máquina e os colegas. (VALENTE, 1999). Este atrativo realmente pode deixar o aluno interessado no conteúdo proposto no jogo colaborando para os interesses educacionais.

Para Oliveira; Costa e Moreira (2001) este atrativo caracteriza uma “motivação pessoal”, pois o lúdico é a essência do jogo informatizado, mas o espírito de competição é despertado do aluno com computador ou colegas. Entretanto, assim como os outros tipos de *softwares*, os jogos educativos também possuem suas desvantagens. Valente afirma que os jogos:

Têm a função de envolver o aprendiz em uma competição e essa mesma competição pode desfavorecer o processo de aprendizagem: por exemplo, dificultando o processo de tomada de consciência do que o aprendiz está fazendo e, com isso, dificultando a depuração e, por conseguinte, a melhora do nível mental (VALENTE, 1999, 81).

Atualmente há jogos que apresentam características similares aos outros tipos de *softwares*. Isto pode minimizar estas desvantagens e trazer as vantagens

em cada um dos tipos *softwares* citados (Jucá, 2011). O autor indica que os jogos podem aprimorar o raciocínio lógico e dedutivo, estimular o cálculo mental, pois podem envolver situações dependentes do tempo de resposta.

3.3. Classificação de *softwares* educacionais em matemática

Na educação de Matemática esta classificação anterior prevalece, mas há outras denominações que aparecem de forma sintetizada, conforme conteúdos e em relação à interação computador-aluno. As denominações mais comuns são: *softwares* algébricos ou folhas de cálculo, *softwares* de geometria dinâmica e *softwares* de representação gráfica.

3.3.1. *Softwares* algébricos ou de folha de cálculo

Os *softwares* algébricos ou folha de cálculo possuem capacidade de manipular, de maneira simbólica, expressões matemáticas e realizar cálculos numéricos. Para Oliveira e Domingos (2008) este tipo de *software* constitui nova maneira de representar e solucionar um problema numérico, reunindo características voltadas para calculadoras, *softwares* de programação, modelagem e simulação (estes dois últimos podem ser um só conforme apresenta-se na subseção 3.2.4).

Além disso, os problemas nos campos da aritmética, da álgebra, da trigonometria, do cálculo, da álgebra linear podem ser resolvidos através do clicar do mouse e até de sistemas de programação. Atualmente, está se tornando comum apresentarem outros tipos de representações, por exemplo, a representação gráfica, simulação e principalmente, linguagem compatíveis com outros *softwares*, ampliando suas potencialidades e maior facilidade no manuseio (OLIVEIRA; DOMINGOS, 2008).

O estudo perante os *softwares* nesta característica tem menor expressão que a investigação entorno de “geometria dinâmica”. Apesar de sua utilização estar centralizada aos níveis superiores de educação o uso dos *softwares* “folha de cálculo” ou “algébricos” nos níveis de escolaridade mais elementares contribuiu na transição da aritmética para a álgebra e como uma ferramenta importante na

resolução de problemas (SANT'ANA; AMARAL; BORBA, 2012). Na Tabela 3 são apresentados alguns *softwares* que pertence a esta categoria.

Tabela 3 - *Softwares* de folha de cálculo ou algébricos.

SOFTWARES	IDIOMA	DISPONIBILIDADE
Derive	Inglês	Comercial (Pago)
WinMat	Inglês	Livre (Gratuito)
Maple	Inglês	Comercial (Pago)
Maxima	Inglês	Livre (Gratuito)
Octave	Inglês	Livre (Gratuito)
VisualMethods	Inglês	Comercial (Pago) e Versão DEMO ⁸
3D-Filmstrip	Inglês	Livre (Gratuito)
Mathematica	Inglês	Comercial (Pago)
MatLab	Inglês	Comercial (Pago)
Scilab	Inglês e português	Livre (Gratuito)
MathCad	Inglês	Comercial (Pago)
MuPAD	Inglês	Comercial (Pago)

Fonte: Elaborado pelo autor.

3.3.2. *Softwares* de geometria dinâmica

É caracterizado pelo seu dinamismo evidenciado em sua capacidade de utilizar, manusear, combinar, visualizar e construir virtualmente objetos geométricos. Dentre suas vantagens este tipo de *software* possibilita o trabalho simultâneo em ambiente geométrico e algébrico (OLIVEIRA; COSTA; MOREIRA, 2001; BORBA; SCUCUGLIA; GADANIDIS, 2014).

Um dos mais conhecidos *softwares* nesta categoria é *Geogebra*, disponível gratuitamente aos usuários e com vários estudos sobre a sua utilização. O *software* contempla vários conteúdos, principalmente a construção geométrica e representação gráfica de funções em planos bidimensionais e tridimensionais.

As potencialidades relacionadas aos *softwares* de geometria dinâmica envolvem: precisão e variedade na construção de objetos geométricos; exploração e

⁸ Versão demo ou de demonstração é uma fração de um produto maior, lançado com a intenção de dar a oportunidade de o produto ser avaliado por possíveis clientes.

descoberta, visualização ou representação mental de objetos geométricos; e avaliação recíproca (ALVES; SOARES, 2003). Na Tabela 4 encontram-se alguns dos *softwares* de Geometria Dinâmica que estão disponíveis aos usuários, idioma e disponibilidade.

Tabela 4 - *Softwares* de Geometria Dinâmica.

SOFTWARES	IDIOMA	DISPONIBILIDADE
Cinderella	Inglês e português	Livre (Gratuito)
Cabri Géomètre II	Inglês e português	Comercial (Pago) e Versão DEMO
Geogebra	Português	Livre (Gratuito)
Dr Geo	Português	Livre (Gratuito)
Construfig 3D	Português	Livre (Gratuito)
Euklid	Inglês	Livre (Gratuito)
Geometria Descritiva	Português	Livre (Gratuito)
The Geometer's Sketchpad	Inglês	Comercial (Pago)
Geoplan	Inglês	Comercial (Pago) e Versão DEMO
Geospace	Inglês	Comercial (Pago) e Versão DEMO
Régua e Compasso	Português	Livre (Gratuito)

Fonte: Elaborado pelo autor.

3.3.3. *Softwares* de representação gráfica

Os *softwares* são, basicamente, programas automatizados para a representação gráfica de funções matemáticas. São programados para auxiliar o processo criativo, agilizando e simplificando o trabalho do usuário. Possuem ênfase na construção de gráficos a partir de funções pré-definidas, onde podem ser inseridas manualmente (DOSCIATI et al., 2004).

Os *softwares* de representação gráfica reconhecem as funções inseridas pelo usuário e associam as características contidas em sua programação. A partir daí, cria, automaticamente, um vetor com todos os valores pertencente ao conjunto domínio e, conseqüentemente, o do conjunto imagem. Gerar um gráfico acaba se tornando fácil em um software com esta característica (DOSCIATI et al., 2004).

Borba; Scucuglia e Gadanidis (2014) afirmam que *softwares* com estas características possibilitam que novos tipos de problemas e atividades matemáticas

sejam descobertos. A sua utilização pedagógica, geralmente, volta-se ao estudo de comportamento de gráficos, controles de parâmetros e, em alguns casos, simulação gráfica. Na Tabela 5 são evidenciados alguns *softwares* que pertencem a categoria.

Tabela 5 - *Softwares* de representação gráfica.

SOFTWARES	IDIOMA	DISPONIBILIDADE
WinPlot	Inglês e português	Livre (Gratuito)
Graphmatica	Inglês	Livre (Gratuito)
CurveExpert	Inglês	Livre (Gratuito)
iGraf	Português	Livre (Gratuito)
Graphequation	Inglês	Livre (Gratuito)
Mathgv	Inglês	Livre (Gratuito)
Modellus	Português	Livre (Gratuito)

Fonte: Elaborado pelo autor.

4. METODOLOGIA

Ao realizar um trabalho de cunho científico o pesquisador segue uma sequência de métodos e procedimentos, visando estruturar a abordagem que assumiu diante de um determinado estudo. Dessa forma, a metodologia busca fundamentar e relacionar os procedimentos com a relevância dos resultados obtidos (MARCONI; LAKATOS, 2003).

É comum as metodologias estarem ligadas a natureza de cada Ciência onde, ao longo do desenvolvimento do conhecimento científico, foram adotando-se práticas específicas de investigação. Para Gil (2002) as metodologias são classificadas conforme modelo conceitual e operativo da pesquisa, ou seja, relacionando os objetivos gerais delimitados e/ou baseadas em nos procedimentos adotados.

Para o presente trabalho adotou-se um posicionamento exploratório diante do *software* Scilab. Por se tratar de um *software* amplamente utilizado no ensino superior, principalmente nas ciências das engenharias, fazia-se necessário identificar o seu caráter pedagógico neste e em outros níveis da educação.

A pesquisa exploratória busca “proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou a constituir hipóteses” (GIL, 2002, p. 41). Dessa forma, caracteriza-se como o estudo preliminar realizado com o intuito adequar instrumentos à realidade que se pretende conhecer como parte de uma pesquisa maior ou para pesquisas futuras. Nesse sentido, a pesquisa exploratória possibilita ao pesquisador a descoberta de principais enfoques e a criação de novas percepções.

Há uma divergência de opiniões sobre a flexibilidade da pesquisa exploratórias, mas, basicamente, assume a forma de pesquisa bibliográfica e/ou de estudo de caso Gil (2002). Para este trabalho optou-se por realizar um levantamento bibliográfico sobre a utilização do Scilab com fins educacionais, buscando informações sobre recursos e conteúdos contemplados no *software*. É essencial, antes de uma aplicação em sala de aula, saber manuseá-lo e conhecê-lo pedagogicamente para que haja uma contribuição significativa para o ensino de matemática.

Segundo Marconi e Lakatos (2003) pesquisas bibliográficas buscam inteirar o pesquisador com tudo o que foi produzido sobre assunto, pois mesmo que haja poucas referências sobre o assunto pesquisado, nenhuma pesquisa hoje começa totalmente do zero. Dessa forma, possibilita identificar o rumo que as pesquisas estão sendo encaminhadas, possibilitando ao pesquisador tratar o tema sob uma nova perspectiva ou com um novo enfoque (olhar totalmente exploratório).

Os dados obtidos receberam um tratamento qualitativo, sendo possível identificar: as principais linhas pesquisas, conteúdos abordados e a relevância do *software* no ensino de Matemática. Na análise qualitativa o pesquisador desenvolve conceitos, ideias e entendimentos a partir de padrões encontrados nos dados (GERHARDT; SILVEIRA, 2009). Apesar de depender de muitos fatores, a pesquisa qualitativa basicamente se resume: a redução e categorização de dados, sua interpretação e a redação do relatório. As autoras também afirmam que:

Os pesquisadores que utilizam os métodos qualitativos buscam explicar o porquê das coisas, exprimindo o que convém ser feito, mas não quantificam os valores e as trocas simbólicas nem se submetem à prova de fatos, pois os dados analisados são não-métricos (suscitados e de interação) e se valem de diferentes abordagens (GERHARDT; SILVEIRA, 2009, p. 32).

Dessa forma, os dados levantados para este trabalho foram organizados conforme a complexidade dos conteúdos utilizados no Scilab. Como alguns conteúdos exigem mais conhecimento sobre conceitos matemático e lógica programação, a forma com a qual é utilizado diferencia-se das outras que exigem menos.

Ao final do trabalho é realiza-se uma comparação entre um conteúdo aplicado com métodos usualmente utilizados em livros didáticos e com o *software* para analisar se há uma discrepância entre os dois. O conteúdo utilizado pertence ao 9º ano do Ensino Fundamental está relacionado à construção gráfica de uma função polinomial, um exemplo com uma função do 1º grau e outro com a de 2º grau.

Outra comparação fez-se entre o Scilab e um *software* de representação gráfica. O conteúdo proposto volta-se a construção gráfica de uma função que possui uma inexistência no domínio. Os *softwares*, baseados em suas características, tratam este tipo de função de forma diferente implicando na forma que interage com o aluno.

Estas comparações baseiam-se em uma análise de conteúdo que constitui uma metodologia de pesquisa usada para descrever e interpretar o conteúdo de toda classe de documentos e textos. Essa análise tem por finalidade a reinterpretação das informações obtidas e a atingir uma compreensão de seus significados num nível que vai além de uma leitura comum (MARCONI; LAKATOS, 2003).

Com isso, busca-se comparar as informações cedidas pelo Scilab com a abordagem adotada nos livros didáticos. Para realizar estas duas comparações foi necessário estudar os comandos do *software* Scilab que contemplavam este conteúdo em específico. Além disso, outros estudos do *software* foram desenvolvidos, tais como: sistema de ajuda, conteúdos disponíveis e respostas de erros.

5. O SOFTWARE SCILAB

Este capítulo está dividido em quatro seções onde apresenta-se a estrutura dos *softwares* Scilab, as produções científicas, uma simulação de aplicação de conteúdos através do *software* e uma breve discussão. A escolha por este *software* está ligada diretamente a forma que o aluno/usuário interage com a sua interface. Em nenhum momento o Scilab, dentre os conteúdos contemplados, fornece diretamente respostas ao usuário e sim o usuário que fornece ao *software*. Dessa forma, constitui uma excelente para expressar e solucionar problemas matemáticos.

5.1. Estrutura do *software*

O Scilab consiste de um *software* livre e de código aberto para computação numérica. Sua programação fornece, basicamente, um terminal matemático interativo ao usuário. Possibilita a criação de algoritmos complexos em poucas linhas de comandos, em comparação com outras linguagens como C, Fortran ou C++ (BARRETO, 2008).

O *software* foi criado no INRIA (*Institut National de Recherche en Informatique et en Automatique*⁹) na França. Atualmente é mantido pela Scilab Enterprises¹⁰, responsável por desenvolver, editar e distribuir o *software*. A empresa fornece serviços profissionais relacionado a formação docente e desenvolvedores; apoio técnico e desenvolvimento de novas plataformas. Na França, cede da empresa, a relação com o sistema de ensino é bem estruturada sendo utilizados na educação secundária e superior (comumente aplicado na Matemática, Ciências das Engenharias e Engenharia de Controle).

Ao elaborar e participar de conferências pelo mundo os gestores e desenvolvedores do Scilab buscando estruturar está relação com outros sistemas de ensino. Por apresentar código aberto, muitos projetos de terceiros estão incorporados a estrutura principal ou estão disponíveis para *download* separadamente. Apesar de ter um grupo de desenvolvedores, a Scilab Enterprises utiliza amplamente os recursos desenvolvidos por seus usuários visando ampliar a

⁹ Instituto Nacional de Pesquisa em Informática e Automação.

¹⁰ Disponível em: <http://www.scilab.org/>.

área de atuação do *software*. Esta ligação da empresa com os usuários auxilia na busca por melhorias e *bugs* (erros) na programação.

A última versão do Scilab 5.5.2 disponível aos usuários apresenta um grande número de funcionalidades ligadas à: Matemática e Simulação; Visualização em 2D e 3D; Otimização; Estatísticas; Sistema de Controle e Análise; Processamento de Sinais; Desenvolvimento de Aplicações; Modelador e simulador de sistemas dinâmicos híbridos – Xcos (BARRETOS, 2008). A Tabela 6 apresenta a composição de cada uma destas funcionalidades.

Tabela 6 - Lista de funcionalidades do Scilab

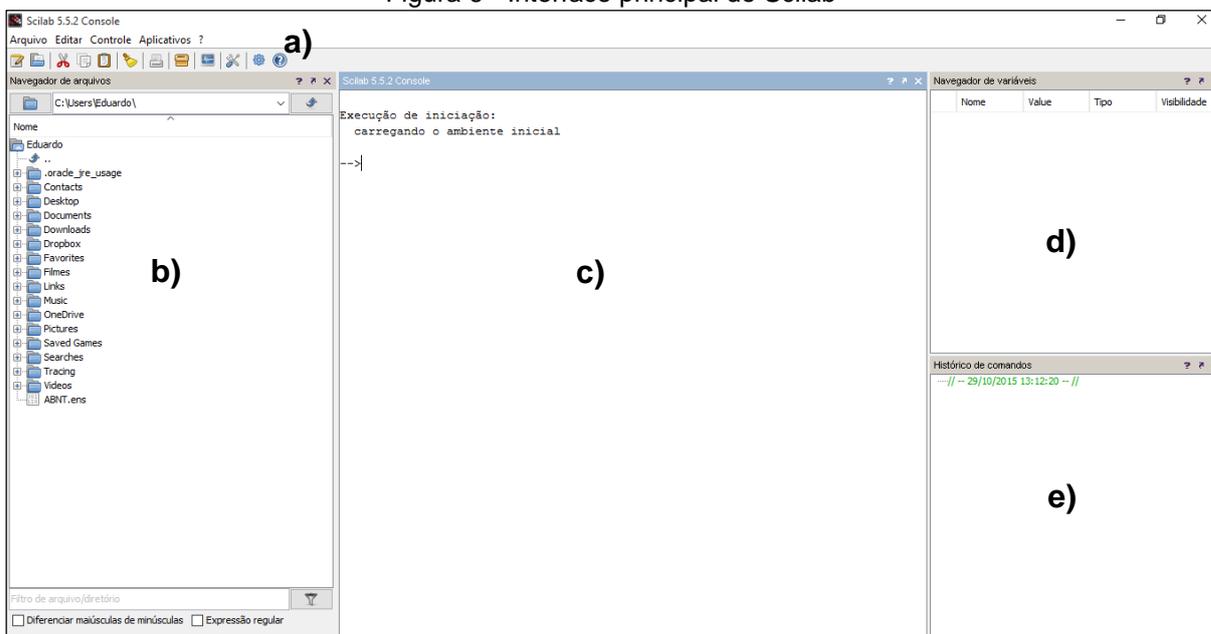
Matemática e Simulação	Funções matemáticas elementares; Álgebra Linear; Matrizes esparsas; Interpolação Polinômios e funções racionais; Simulação: sistemas explícitos e implícitos de equações diferenciais;
Visualização em 2D e 3D	Linhas; Gráfico de setores; Histogramas; Superfícies; Animações; LaTeX/MathML; Gráficos de exportação em vários formatos: PNG, PPM, EMF, EPS, FIG, PDF;
Otimização	Não-linear; Quadrática; Otimização por mínimos quadrados; Programação semidefinida; Algoritmos genéticos; Reconhecimento simulado; Lineares desigualdades matriciais;
Estatísticas	Estatísticas descritivas; Distribuições de probabilidade; Modelagem linear e não-linear;
Sistema de Controle e Análise	Função de transferência; Controle clássico e robusto; Análise de estabilidade; Análise de espaço do estado design do sistema de controle;
Processamento de Sinais	Geração de sinal; Janelas de dados; Estimativa da densidade espectral de potência; Projeto de filtros digitais FIR e IIR; Design de filtro analógico;
Desenvolvimento de Aplicações	Estruturas avançadas de dados e tipos de dados definidos pelo usuário; Interface com o Fortran, C, C ++, Java; Construção dinâmica e ligação; ATOMS (gerenciamento automático de módulos para Scilab); <i>Framework</i> de testes;
Modelador e simulador de sistemas dinâmicos híbridos - Xcos	Paletas e Blocos padrão; Modelo de construção e Edição; Modelos Personalização; Simulação

Fonte: Elaborado pelo autor.

Todas estas funcionalidades podem ser acessadas através de comandos inseridos no *console* do *software* (Vide Anexo A – Lista de comandos do Scilab). A

interface principal do Scilab apresenta os seguintes itens: a) menu e barras de ferramentas; b) navegador de arquivos; c) *console*; d) navegador de variáveis; e) e histórico de comandos. A Figura 6 mostra como cada um destes itens é posicionado na interface.

Figura 6 - Interface principal do Scilab



Fonte: Elaborado pelo autor.

O Scilab também possui: uma janela de ajuda que contém instruções e exemplos para inserir comandos; uma janela gráfica que permite visualizar e editar gráficos; uma janela para o navegador de *paleta* – Xcos; uma janela para visualização de simulações; uma janela de Módulos ATOMS que permite instalar módulos externos e adicionar novas funcionalidades para Scilab em vários domínios de aplicações (desenvolvido pela equipe Scilab Enterprises ou pela comunidade), vide Anexo B.

5.2. Produções acerca do Scilab

Na busca por trabalhos que evidenciasse a utilização do Scilab como ferramenta de ensino-aprendizagem foi delimitada as produções encontradas no período de 2004 a 2015. A justificativa por este período está ligada a realização da primeira conferência internacional sobre o Scilab realizada em 2004. Os tópicos abordados na conferência foram: Simulação; Sistemas dinâmicos; Processamento

de imagens e de sinais; Controle; Modelagem e identificação; Otimização; Métodos computacionais; Aplicações em engenharia; Ensino. As produções acadêmicas ligadas a utilização do Scilab no ensino podem ter sido impulsionadas devido à presença de um tópico em específico na conferência organizada pelos próprios desenvolvedores e gestores nos *softwares*.

Os trabalhos selecionados que tratam sobre o tema mostram que as pesquisas relacionadas a utilização do Scilab no ensino-aprendizagem estão centralizadas no ensino superior, mais especificamente nas áreas das engenharias. Os trabalhos de Freitas (2004) e Almeida Junior (2012) evidenciam, respectivamente, a utilização do Scilab para a resolução de sistemas lineares e operadores de Controle Automático de um sistema elétrico de potência no curso de Engenharia Elétrica.

As potencialidades evidenciadas no Scilab pode torna-lo um substituto ao MatLab, tido como a elite entre os *softwares* algébricos utilizados tanto no meio acadêmico como no mercado profissional. As semelhanças apresentadas entre os *softwares* envolvendo o tratamento de variáveis numéricas e simbólicas, simulação, apresentação de diagramas e a muitas das funções pré-definidas realçam a utilização do Scilab devido à disponibilidade gratuita aos usuários (FREITAS, 2004; ALMEIDA JUNIOR, 2012).

Nos trabalhos de Mariani; Preto e Guedes (2005), Amaral; Leite e Silva (2013), Mota (2011), Silva (2013) e Silva e Cunha (2006) a utilização do Scilab está voltada para a disciplina de Cálculo Numérico como os alunos dos cursos de Matemática e Engenharia Civil, Química e Elétrica.

Silva (2013) mostra que a utilização de tecnologias computacionais no ensino de Cálculo Numérico não tem por finalidade o computador realizar os cálculos para o aluno, mas sim de trazer novas situações matemáticas ao contexto educacional e uma nova maneira de expressar o conhecimento matemático. Segundo o autor, o Scilab é somente uma ferramenta utilizada para processar uma determinada informação escrita pelo aluno.

A linguagem de programação utilizada no Scilab é de alto nível, mas não exige do usuário técnicas avançadas de programação e estruturas de dados. A capacidade de resolver problemas numéricos e controlar parâmetros está ligada ao

nível de conhecimento matemático que o usuário consegue expor através do *software* (MARIANI; PRETO; GUEDES, 2005; SILVA; CUNHA, 2006; MOTA, 2011). Dessa forma, a utilização pode envolver de níveis mais elementares a níveis complexos.

O usuário pode elevar o grau de complexidade de utilização do Scilab com caixas de ferramentas (Toolbox) que permitem criar programas para resolver um caso ou um problema específico (AMARAL; LEITE; SILVA, 2013). Isto pode requerer do usuário um conhecimento de programação e conceitos matemáticos mais avançados, dependendo o conhecimento que se deseja atingir.

Nos trabalhos de Souza e Dandolini (2009), Gonçalves (2011) e Bertoldi (2012) este potencial é explorado. A modelagem e a simulação oferecida pelo Scilab através de uma programação construída pelos usuários exploram a resolução de problemas envolvendo parâmetros e sistema otimização numérica proporcionada pelo software na área de Engenharia Civil e Elétrica.

Alguns *softwares*, por serem menos abrangem ou com foco em específico, podem não apresentar uma extensa biblioteca de funções pré-definida e uma abordagem ampla da linguagem de programação. Porém, Gonçalves (2011) e Bertoldi (2012) mostram que programas e simulações criadas no Scilab possuem comunicação com outros *softwares*, ou seja, o usuário pode utiliza-lo paralelamente com outros *softwares*.

Os autores evidenciaram que a comunicação do Scilab com o MatLab, COCO (*software* de simulação) e Excel proporcionou ganhos nas funcionalidades propostas no *software*, expandindo-se os limites em termos de conteúdo. Está aplicabilidade do Scilab na área educacional e profissional se dá justamente pela amplitude de ferramentas disponíveis e pela versatilidade com outros *softwares* (SOUZA; DANDOLINI, 2009).

Os trabalhos de Rodrigues Longo; Lamblém e Duarte (2012) e Trindade (2013) evidenciaram a utilização do Scilab e de outros *softwares* com curso de Licenciatura em Matemática contribuído para o ensino de determinante matricial e análise numérica. Os conteúdos ao serem inseridos no Scilab são baseados em uma sequência de comando gerada pelo usuário que seguem tanto os aspectos intrínsecos do *software* quanto as leis que regem a matemática (RODRIGUES LONGO; LAMBLÉM; DUARTE, 2012; TRINDADE, 2013).

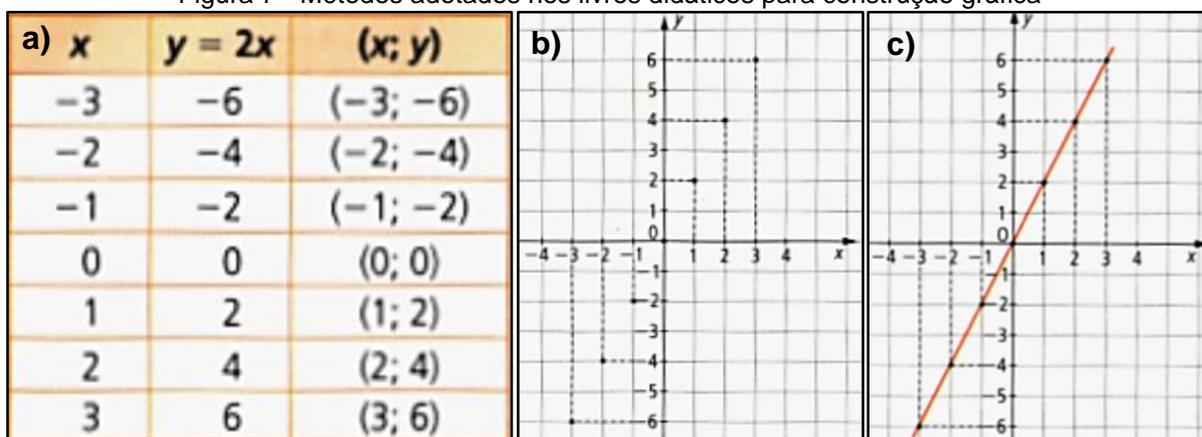
Os conteúdos com menor grau de complexidade são facilmente inseridos no *software*, possuindo uma grande semelhança com atividades desenvolvidas com papel e lápis. Os dados são fundamentalmente vetores ou matrizes de números, lembrando as “folhas de cálculo” (TRINDADE, 2013). Um ponto defendido por Rodrigues Longo; Lamblém e Duarte (2012) está na relação de tentativa e erro oferecida pelo *software*. Como o *software* possui forma coesa resposta para erros cometidos, o usuário pode rapidamente identificar o erro que cometeu, se foi ligado as leis matemáticas ou um erro de comando.

5.3. Comparando uma aplicação do Scilab

Ao realizar um estudo sobre as potencialidades envolvendo os conteúdos contemplados no *software* Scilab deve-se avalia-lo em diversos aspectos. De acordo com Gladcheff (2001), Gomes et al. (2002), Ramos e Mendonça (1991), Martins (2002) e Rocha e Campos (1993), na seção 3.1, um dos pontos que deve ser avaliado é se as atividades propostas podem desenvolvidas com ou sem o *software*, verificando-se há proximidade entre as duas.

Tomando como exemplo o conteúdo envolvendo a construção gráfica de uma determinada função pode-se fazer uma breve análise avaliativa. É um conteúdo em específico, já trazido desde o ensino fundamental nos livros didáticos. A Figura 7 traz um exemplo de como a construção gráfica de Função do 1º Grau deve proceder conforme o método abordado nos livros didáticos do Ensino Fundamental.

Figura 7 - Métodos adotados nos livros didáticos para construção gráfica

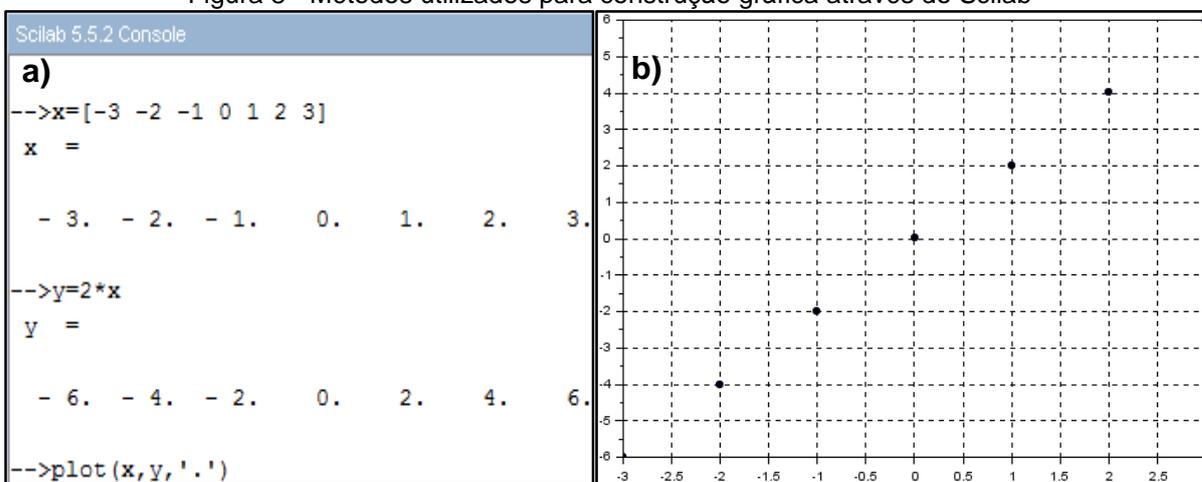


Fonte: Extraído de ANDRINI; VASCOCELLOS (2002, p. 115).

O método utilizado segue as seguintes etapas: a) o aluno deve, primeiramente, construir uma tabela e, na primeira coluna, atribuir valores à variável independente, neste caso x . O segundo passo é substituir esses valores na função dada por $y = 2x$ e verificar cada uma das respostas. Com isso, o aluno terá formado uma sequência de pares ordenados; b) os valores respectivos de x e y serão utilizados como coordenadas para construir o gráfico da função, inserindo pontos discretos no plano cartesiano; c) os pontos devem ser ligados para obter o gráfico da função $y = 2x$.

Utilizando o mesmo exemplo, porém resolvidos através do *software* Scilab, o aluno deverá aplicar as seguintes etapas: a) especificar ao *software* quais os valores que serão atribuídos a variável independente e o *software* confirmará a escolha. O segundo passo será especificar a função que se deseja trabalhar e o *software* fará a relação com o conjunto domínio e imitará o conjunto imagem; b) inserir o comando `plot(x,y, '.')` que irá associar as coordenadas obtidas e colocá-las no plano cartesiano gerando pontos discretos. A Figura 8 mostra estas etapas inseridas no Scilab.

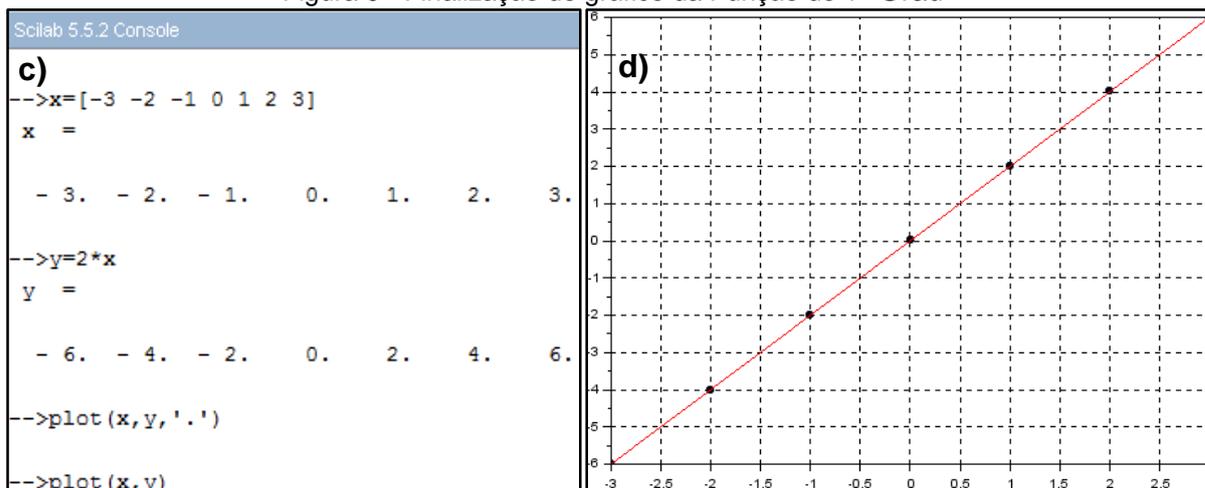
Figura 8 - Métodos utilizados para construção gráfica através do Scilab



Fonte: Elaborado pelo autor.

Pode-se perceber que na construção do gráfico da função realizada através do *software* Scilab não houve um distanciamento com atividade expressa no livro didático. As etapas a serem seguidas são semelhantes, restando apenas ligar os pontos para finalizar a representação gráfica. Para isto, o aluno deverá aplicar: c) o comando que conecta um ponto ao outro dado por `plot(x,y)`, assim o gráfico será finalizado d). A Figura 9 mostra o desenvolvimento desta última etapa.

Figura 9 - Finalização do gráfico da Função do 1º Grau



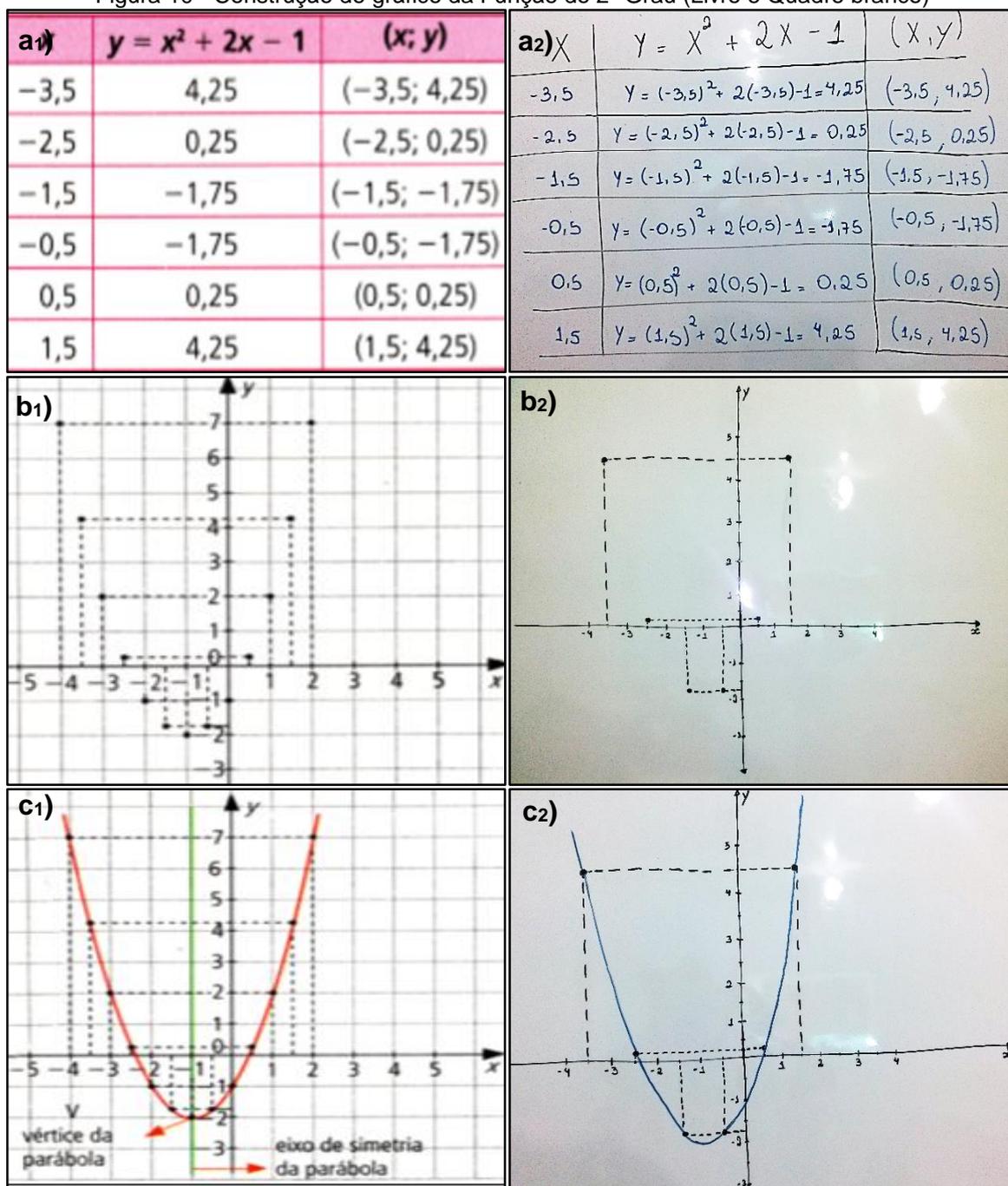
Fonte: Elaborado pelo autor.

Esta comparação, apesar de ser bem simples, é uma maneira de avaliar se a atividade proposta pode ser resolvida com ou sem o *software* e como um determinado conteúdo pode ser desenvolvido. Com esta atividade desenvolvida no livro didático e no Scilab pode-se verificar há uma proximidade entre os métodos utilizados. Este método de atribuição e substituição de valores é o mais utilizado para construir representação gráfica e pode ser aplicado em diversos tipos de funções matemática.

O gráfico da função anterior assumiu um formato geométrico que se assemelha a uma reta. Em algumas funções o comportamento do gráfico assume outras formas peculiares, por exemplo, a Função 2º Grau possui um gráfico que recebe o nome de parábola. Sua concavidade, abertura e posicionamento no plano cartesiano estão relacionados aos valores de seus coeficientes. Demonstrar a suavização da curva nesta função, através da construção gráfica, pode-se tornar algo exaustivo para fazer com lápis e papel.

Ao fazer a construção gráfica no quadro, neste caso branco, o professor acaba tomando como base os mesmos pontos do livro didático e faz a suavização da curva manualmente. Isto acontece devido à quantidade excessiva de pontos que devem ser inseridos para visualizar este comportamento. As etapas adotadas para construir o gráfico de uma Função do 2º Grau seguem as mesmas etapas citadas no exemplo anterior. A Figura 10 mostra a construção trazida no livro didático do ensino fundamental e a resolução no quadro branco.

Figura 10 - Construção do gráfico da Função do 2º Grau (Livro e Quadro branco)

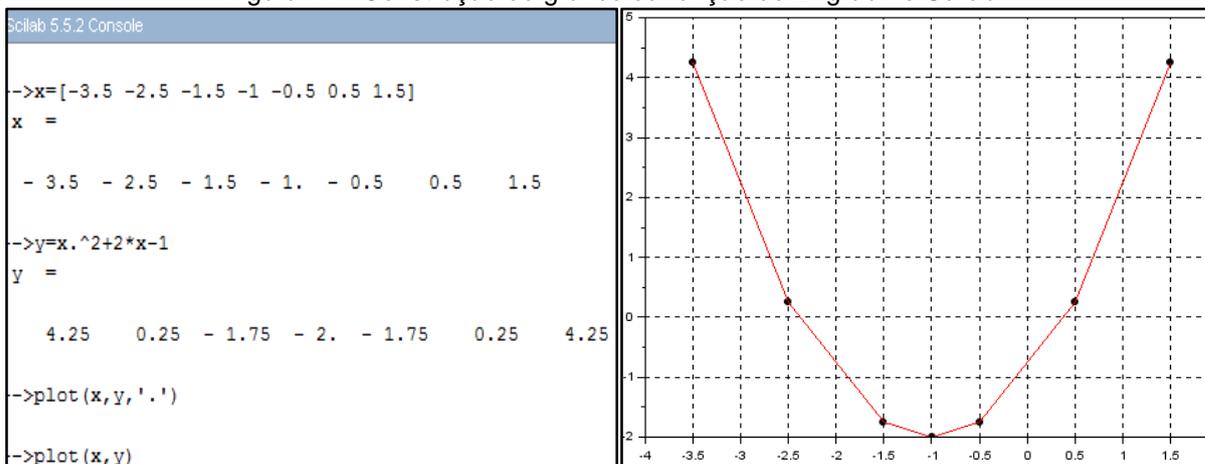


Fonte: Extraído de ANDRINI; VASCOCELLOS (2002, p. 115); Elaborado pelo autor.

A resolução utilizada no livro didático e no quadro branco apresentam as seguintes etapas tem-se: a1) e a2) tem-se a tabelas de atribuição de valores a variável independente e obtenção dos pares ordenados; b1) e b2) pontos colocados discretamente no plano cartesiano; c1) e c2) os pontos são ligados para gerar o gráfico. Ao resolver no quadro branco o professor acaba suavizando a curva ao ligar os pontos. Ao desenvolver a atividade no Scilab o aluno poderá se deparar com uma

situação que poderá instigá-lo. O posicionamento do professor será fundamental para auxiliar o aluno a depurar as informações obtidas através do *software*. Na figura 11 é feita a demonstração a resolução através do Scilab.

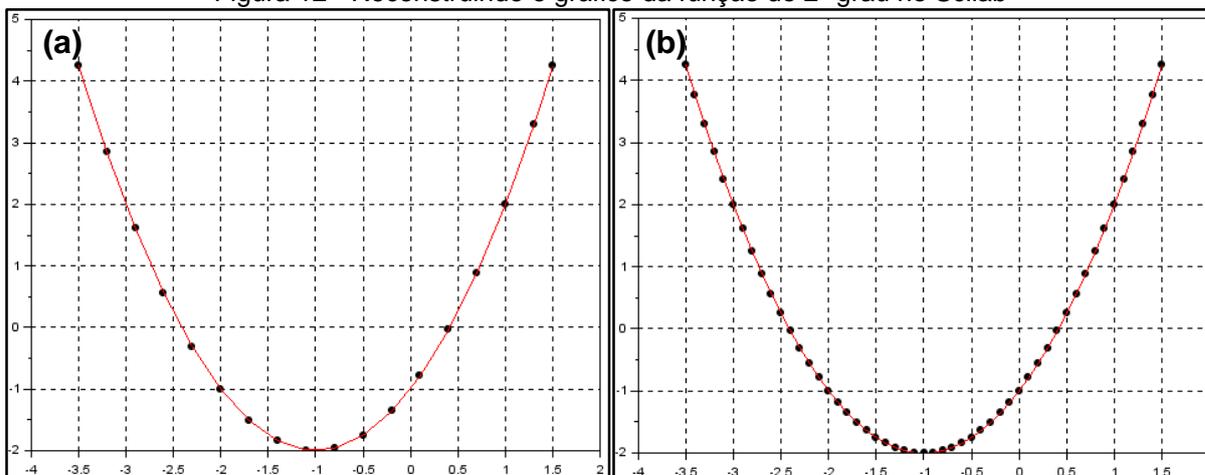
Figura 11 - Construção do gráfico da função do 2º grau no Scilab



Fonte: Elaborado pelo autor.

Por que o gráfico não assumiu este visual perfeitamente suavizado na curva como a resolução do livro ou no quadro branco? A resposta para esta pergunta está na quantidade e no intervalo entre os pontos formados. Os valores atribuídos a variável x , neste caso, foram interpolados manualmente em 0,5 ou 1,0, os pontos formados foram ligados entre si por seguimento de retas. E se atribuir valores com interpolações cada vez menores? Na Figura 12 é feita construção gráfica da função com valores igualmente interpolados cada vez menores.

Figura 12 - Reconstruindo o gráfico da função do 2º grau no Scilab



Fonte: Elaborado pelo autor.

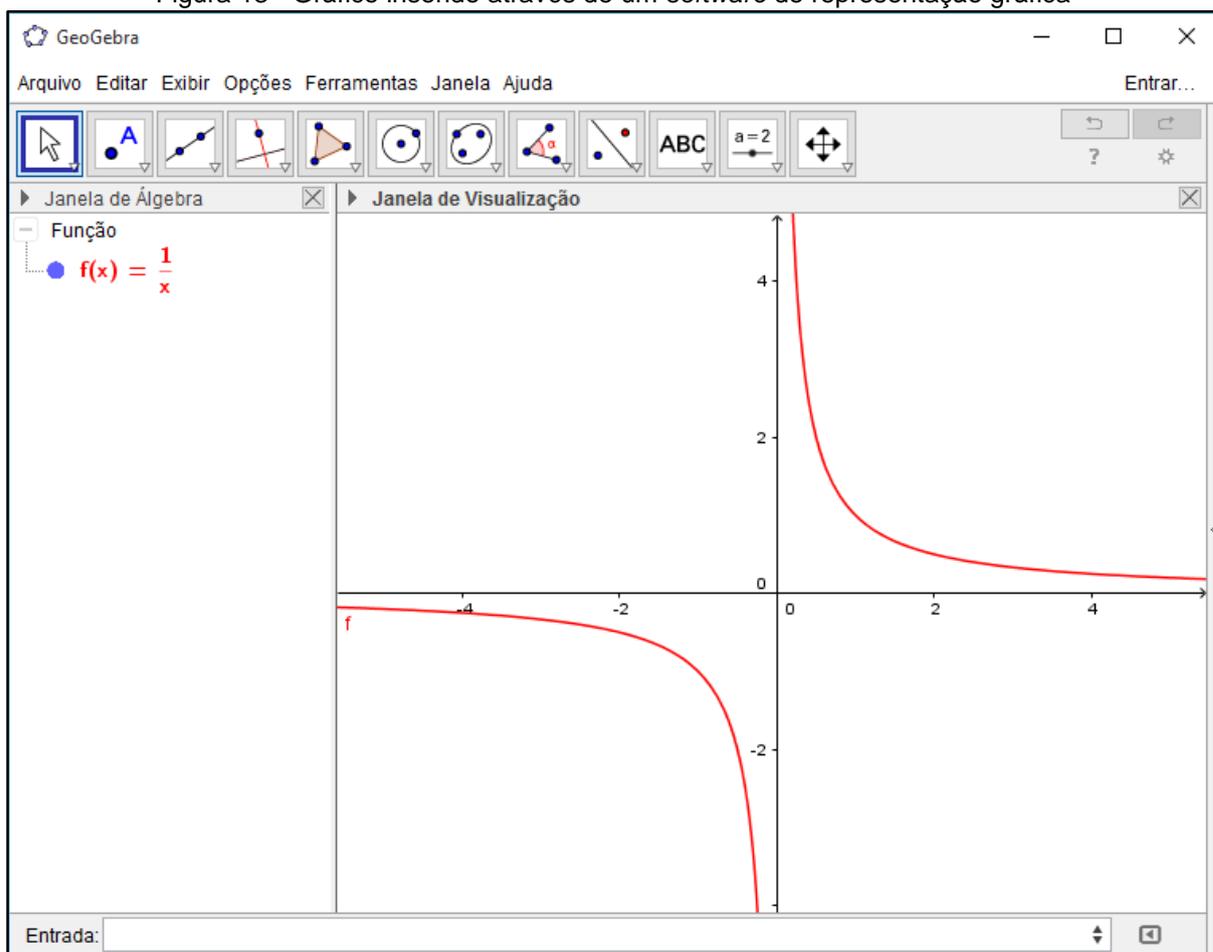
O gráfico da Função do 2º Grau (a) foi construído utilizando a interpolação fornecida no Scilab. O comando $x=-3.5:0.3:1.5$ atribuiu todos os valores entre -3,5 e 1,5 interpolados igualmente em 0,3. Isso originou mais pontos e a distância entre eles acabou diminuído começando a suavizar a curva. No gráfico (b) os valores de x entre -3,5 e 2,5 foram interpolados igualmente em 0,1, os pontos ficaram ainda mais próximos e, ao liga-los uns com os outros, o gráfico se assemelhou com construído no livro didático. Isto acontece devido os seguimentos ficaram cada vez menores, tornando quase imperceptíveis. Para inserir uma grande quantidade de pontos manualmente (lápiz e papel) e obter uma representação semelhante, a atividade demandaria de tempo e poderia ser tornar exaustiva durante a resolução.

O conteúdo de construção gráfica continua o mesmo e o aluno ainda deve ser as mesmas etapas citadas no livro didático. Entretanto, o Scilab possui outras maneiras de contemplar estas etapas. Comando com este, e outros, maximizam a criação de pontos e minimizam o tempo para digitá-los. Ferramentas como esta possibilitam novas formas de resolução para problemas matemáticos e ampliam as situações pedagógicas (BORBA; SCUCUGLIA; GADANIDIS, 2014).

Esta comparação avaliativa também pode ser feita com outros *softwares* já que cada um é desenvolvido sob finalidades e objetivos específicos (GLADCHEFF, 2001). Ao realizar a escolha entre determinados *softwares* o docente deve ater-se a conteúdo que vai aplicar. Se há *softwares* que podem fazer representações gráficas rapidamente, bastando apenas inserir a função que o gráfico já aparece pronto, por que não utilizá-lo para ensina construção gráfica?

Para Borba; Scucuglia e Gadanidis (2014) ao remente-se os *softwares* à conteúdos projeções gráfica de funções deve-se levar em consideração o que se está ensinado, a construir gráfico ou a estudar comportamentos de gráficos. Segundo o autor, *softwares* de representação gráfica, como citados em seções anteriores, costumam fornecer gráficos prontos.

Neste caso, o aluno não participa integralmente na construção do gráfico, limitando-se a inserir a função no *software* que se está trabalhando. Como exemplo, pode-se aplicar uma função que não possui solução para um determinado valor do domínio em um *software* de representação gráfica. A Figura 13 mostra a função $f(x)=1/x$ em um *software* de representação gráfica.

Figura 13 - Gráfico inserido através de um *software* de representação gráfica

Fonte: Elaborado pelo autor.

O gráfico da função foi inserido perfeitamente, mas todas as etapas debatidas nos exemplos anteriores em relação a construção de gráficos foram resolvidas pelo *software* de representação, restando ao aluno somente inserir a função que se deseja trabalhar. Dessa forma, *softwares* com estas características costumam serem utilizados para estudo de comportamento de gráfico, onde o aluno já sabe construir gráfico e está em um conteúdo com outra abordagem didática (BORBA; SCUCUGLIA; GADANIDIS, 2014).

Para construir o gráfico no Scilab desta e outras funções que possuem indeterminação no domínio o aluno deverá compreender claramente as definições que salientam este conteúdo. Como o *software* está baseado em programações que seguem a rigor as leis matemática, caso o aluno tenha algum equívoco com o conteúdo matemático ao inserir as informações o *software* ele imitará uma resposta

sobre o erro. A Figura 14 evidencia um erro que pode ser cometido por um aluno em relação ao conteúdo matemático.

Figura 14 - Informação de erro do Scilab

```

Scilab 5.5.2 Console
-->x=[-3 -2 -1 0 1 2 3]
x =
- 3.  - 2.  - 1.   0.   1.   2.   3.

-->fx=1 ./x
      !--error 27
Divisão por zero...

-->

```

Fonte: Elaborado pelo autor.

Ao atribuir valores para a variável x , o aluno pode cometer o erro de inserir um valor que não pertence ao domínio da função. A resposta fornecida pelo Scilab sobre o erro cometido é, evidentemente, precisa. Dentre os valores atribuídos pelo aluno está o zero, ao substituí-lo na função, dada por $f(x) = 1/x$, torna-se impossível o *software* realizar a operação. “Divisão por zero” não possui solução nas leis matemáticas.

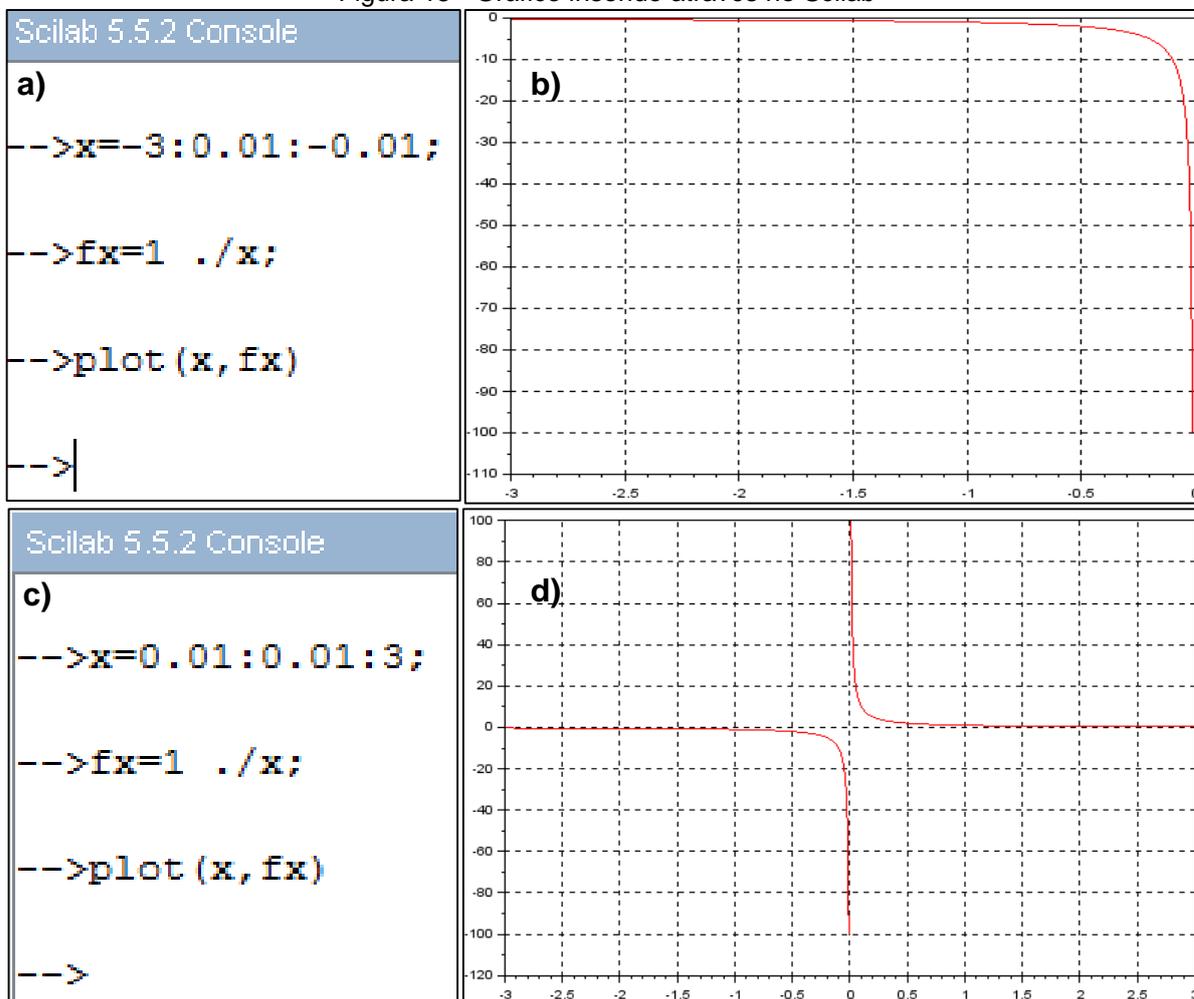
Ao deparar com tal resposta de erro, Valente (1999) afirma que o aluno pode realizar o processo de depuração das informações fornecidas pelo *software*, visando identificar onde está o erro. Este processo só terá efeito se o aluno possui clareza no sobre o conteúdo trabalhado.

Identificando o erro deve-se encontrar uma alternativa para prosseguir com a atividade, pois o erro não foi do *software*. Caso apresente dificuldade ou se sinta desencorajado o professor deve auxiliar o aluno, gerando novos ciclos de depuração de informação.

Sabendo que o único valor que não pertence ao domínio da função é zero uma das opções que o aluno pode encontrar através do Scilab é construir o gráfico da função por partes, atribuindo valores a esquerda do zero, suficientemente próximos, mas não igual a zero e atribuir valores a direita do zero, suficientemente próximos, mas não igual à zero. Nesse ponto o aluno também estará, de certa

forma, sendo iniciado no conceito limite e/ou ponto aberto e ponto fechado. A Figura 15 mostra como poderia ser construir o gráfico da função $f(x) = 1/x$ no Scilab.

Figura 15 - Gráfico inserido através no Scilab



Fonte: Elaborado pelo autor.

Os valores atribuídos valores a esquerda do zero (a), neste caso, estão entre -3 e -0,01 interpolados em 0,01¹¹; (b) ao inserir o comando para o construir o gráfico o aluno deve estar ciente que somente uma parte da função foi representada. Os valores atribuídos direita do zero (c) estão entre 0,01 e 3, interpolados em 0,01; (d) o aluno ao construir o restante do gráfico o aluno poderá compreender o porquê de assumir tal formato.

Apesar de ser um exemplo simples, pode-se perceber que a interação fornecida pelo Scilab é diferente da adotada pelo *software* de representação gráfica.

¹¹ O “ponto e vírgula” no final de cada operação é um comando que suprime a visualização dos valores. Como os números foram interpolados em 0,01, vários valores seriam obtidos e não viável colocá-los em imagens.

O posicionamento do professor é essencial para identificar as potencialidades técnicas e pedagógicas de cada um. Para Gomes et al. (2002) saber avalia-los possibilita conhecer os recursos disponíveis em cada um e criar um relação com o conteúdo a ser aplicado.

5.4. Discussões

Apesar do Scilab estar sendo mais utilizado no Ensino Superior a comparação realizada mostra que o *software* contempla conteúdos presentes tanto nos anos finais do Ensino Fundamental quanto Ensino Médio. Com isso, este *software* pode, além de contribuir em determinados conteúdo, compor uma ferramenta que poderá ser utilizada para uma formação acadêmica posterior.

Sua base de programação fornece facilidade ao manusear e a composição vetorial/matricial da interface numérica coloca o aluno em uma posição em que o conhecimento prévio matemático será de fundamental importância. O Scilab segue fielmente as leis e operações matemáticas, com isso o aluno não conseguirá utilizá-lo se não as conhecer.

A interface do *software* é detalhada em passos semelhante à resolução manual. A saída de cada método visa auxiliar o estudante na resolução de exercícios, pois o mesmo pode conferir em cada passo o andamento da resolução. O aluno deve inicialmente fazer a identificação dos dados de entrada do problema. Posteriormente, poderá acompanhar as informações intermediárias e verificar a saída final (resposta para o problema inicial). Dessa forma, pode-se resumir que o Scilab não é um software que ensinará tudo o que o aluno precisa, mas sim, o aluno que construirá e estruturará o conhecimento matemático por meio Scilab.

6. CONSIDERAÇÕES FINAIS

A partir do estudo pode-se compreender que os *softwares* educacionais são classificados com base nas singularidades, formatos ergonômicos, conteúdos e aplicações observadas no processo avaliativo. As classificações incluem: Tutoriais, Exercícios e Prática, Programação, Simulação e Jogos, podendo existir outras. A linguagem e interação propiciadas é diferente quando comparado uns com os outros. Com isso, entender como são classificados ajuda o docente na hora de escolher qual é o mais apropriado para uma determinada situação.

Referindo-se a classificação dos *softwares* no ensino de matemática pode-se perceber que os métodos utilizados prevalecem, mas algumas categorias ganharam espaço pela forma de interação proporcionada ao usuário, apresentando-se com: *Softwares* de Geometria Dinâmica; *Softwares* Algébricos ou Folha de Cálculo e *Softwares* de Representação Gráfica. Cada categoria possui seus conteúdos e objetivos específicos que buscam atingir.

A busca por tornar estes *softwares* mais abrangentes levou os desenvolvedores a unir e compartilhar algumas ferramentas entre as categorias. Ou seja, *softwares* de geometria dinâmica, por exemplo, podem apresentar elementos relacionados aos *softwares* de representação gráfica; *softwares* algébricos com capacidade de gerar gráfico com extrema qualidade e comunicar-se com outros *softwares*. Isto possibilitou ampliar as potencialidades dos *softwares*.

Ao conhecer os tipos de *software* matemáticos existentes pode-se determinar que o Scilab pertence a categoria do *software* algébricos ou folha de cálculo. Os recursos e conteúdos encontrado no *software* estão relacionados matemática e simulação, visualização em 2D e 3D, otimização, estatísticas, sistema de controle e análise, processamento de sinais, desenvolvimento de aplicações, modelador e simulador de sistemas dinâmicos híbridos – Xcos.

Apesar de possuir recursos e conteúdos passíveis de serem utilizados em níveis de ensino mais elementares, mostrou-se que a sua utilização está focada ao nível superior, com grande ênfase nas Ciências das Engenharias. A amplitude do Scilab e a versatilidade de comunicação e interação com outros *softwares* pode

colocá-lo em um status de comparação com os *softwares* de caráter comercial, levando vantagem principalmente pela disponibilidade gratuita aos usuários.

Ao realizar a comparação do Scilab com métodos adotados nos livros didáticos e em outros *softwares* percebe-se porquê *softwares* algébricos são muitas vezes chamados de Folha de Cálculo. A semelhança entre a resolução realizada papel e lápis com a resolução por meio Scilab está na interação proporcionada. Isto acontece devido *software* possuir um console em branco, esperando receber comando atribuídos pelo aluno.

Na comparação com um *software* de representação gráfica o intuito não foi expressar qual é mais fácil de ser utilizado ou qual é o mais completo, mas qual a relevância que deve ser observada em relação ao conteúdo aplicado. Como o conteúdo envolvia construção de gráficos de determinadas funções, espera-se que o aluno compreenda os procedimentos adotados para construir um determinado gráfico.

O Scilab possui, basicamente, as mesmas etapas apresentadas no livro didático e o aluno participa inteiramente no processo de construção. Já os *softwares* de representação gráfica fornecem gráficos prontos e com pouca participação do aluno, sendo mais utilizados para estudar comportamentos gráficos das funções do que, propriamente, para ensinar a construir gráfico. Dessa forma, os dois *softwares* apresentaram grandes potencialidades, mas possuem utilidades com focos diferentes.

O desconhecimento do processo avaliativo e dos potenciais dos diferentes tipos de *softwares* pode implicar em escolhas equivocadas e resultar em experiências desagradáveis com esta ferramenta.

Compreende-se com este trabalho que a utilização dos *softwares* matemáticos e o processo metodológico a serem aplicados no processo de ensino-aprendizagem para que possa realmente colaborar para o ensino de matemática. O docente deve estar ciente que o manusear do *software* não contribui inteiramente à aprendizagem. Dentre as habilidades exigidas para estruturar a aplicação deste recurso metodológico está o saber avaliar os *softwares*.

Dessa forma, identifica-se que a avaliação entorno dos *softwares* leva em consideração aspectos técnicos e pedagógicos que devem ser analisados antes de

uma aplicação em sala de aula, relacionando-os com os conteúdos e com os objetivos que pretende alcançar. Esta avaliação é importante também para possa conhecer os tipos de *softwares* e suas especificidades.

Como sugestões para pesquisas em trabalhos futuros pode-se citar:

- Identificar razões para os *softwares* algébricos ou folha de cálculo serem pouco utilizado em níveis mais elementares.
- Estudo de campo com aplicações do Scilab em diferentes níveis da educação.
- Explorar os conceitos de Álgebra Linear e Geometria Analítica, pois o *software* possui comandos para trabalhar matrizes e sistemas.

REFERÊNCIAS

ALMEIDA JUNIOR, D. A. Aplicação do software ScicosLab/Scilab no controle automático de geração de sistemas de potência explorando o problema de despacho econômico. Dissertação (Mestrado) – Universidade Federal de Uberlândia, Uberlândia, São Paulo. 2012.

ALMEIDA, M. E. B. Tecnologias na Educação: dos caminhos trilhados aos atuais desafios. **Boletim de Educação Matemática**, v. 21, n. 29, p. 99-129, 2008.

ALVES, G. d. S.; SOARES, A. B. Geometria Dinâmica: um estudo de seus recursos, potencialidades e limitações através do software Tabulae. **In: Anais do Workshop de Informática na Escola**, p.175-186, 2003.

AMARAL, T. R.; LEITE, N. G.; SILVA, A. O. O ENSINO DE CALCULO NUMÉRICO UTILIZANDO O SCILAB. **In: VI Congresso Internacional de Ensino de Matemática**, Canoas - RS, 2013.

ANDRINI, Á.; VASCONCELLOS, M. J. **Praticando matemática, 9º ano**. 3 ed. São Paulo: Editora do Brasil, 2002.

BARRETO, L. S. **Iniciação ao Scilab**. 2 ed. 2008.

BERTOLDI, O. J. Investigação de estratégias de otimização de plantas virtuais usando os softwares COCO, Scilab e Excel. Dissertação (Mestrado) – Universidade Federal de Uberlândia, 2012.

BORBA, M. O computador é a solução: mas qual é o problema. **In: SEVERINO, A. J. e FAZENDA, I. C. A. (Org.). Formação Docente: Rupturas e Possibilidades**. Campinas, SP: Papirus, p.151-162, 2002.

BORBA, M. C. Tecnologias informáticas na Educação Matemática e reorganização do pensamento. **In: BORBA, M. C. (Org.). Pesquisa em educação matemática: concepções e perspectivas**. São Paulo: UNESP, p.285-295, 1999.

BORBA, M. C.; PENTEADO, M. G. Pesquisas em informática e educação matemática. **Educação em Revista**, v. 36, p. 239-253, 2002.

BORBA, M. C.; SCUCUGLIA, R.; GADANIDIS, G. **Fases das tecnologias digitais em Educação Matemática**. São Paulo: Autêntica, 2014.

BRASIL. **Parâmetros Curriculares Nacionais, matemática**. Secretaria de Educação. Brasília 1998.

_____.CNE, MEC. **CP–Resolução nº 02**. Define as Diretrizes Curriculares Nacionais para a formação inicial em nível superior (cursos de licenciatura, cursos de formação pedagógica para graduados e cursos de segunda licenciatura) e para a formação continuada. Brasília: MEC, 2015.

COLL, C. Piaget, o construtivismo e a educação escolar: onde está o fio condutor. **Substratum: Temas Fundamentais em Psicologia e Educação**, v. 1, n. 1, p. 145-164, 1997.

DOSCIATI, A. et al. SOFTWARES LIVRES POTENCIAIS PARA O ENSINO DE MATEMÁTICA. 2004.

FREITAS, F. D. Utilização do software scilab como ferramenta computacional em substituição ao Matlab—Aplicação em sistemas com representação linear. **In**: COBENGE – Congresso Brasileiro de Ensino de Engenharia, 2004. Brasília.

GERHARDT, T. E.; SILVEIRA, D. T. **Métodos de pesquisa**. Porto Alegre: Editora da UFRGS, 2009.

GIL, A. C. **Como elaborar projetos de pesquisa**. 4 ed. São Paulo: Atlas, 2002.

GLADCHEFF, A. Um Instrumento de Avaliação da Qualidade para Software Educacional de Matemática. Dissertação (Mestrado) – Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (ICMC-USP): São paulo 2001.

GLADCHEFF, A. P.; ZUFFI, E. M. Um instrumento para avaliação da qualidade de softwares educacionais de matemática para o ensino fundamental. **In**: Congresso da Sociedade Brasileira de Computação e VII WORKSHOP DE INFORMÁTICA NA ESCOLA. Fortaleza, CE, 2001.

GOMES, A. S. et al. Avaliação de software educativo para o ensino de matemática. **In**: WIE 2002 Workshop Brasileiro de Informática Educativa. Florianópolis: SBC, 2002.

GONÇALVES, R. V. Projeto auxiliado por computador de processos industriais sustentáveis usando os softwares COCO e Scilab. Dissertação (Mestrado) – Universidade Federal de Uberlândia, Uberlândia - SP. 2011.

JUCÁ, S. C. S. A relevância dos softwares educativos na educação profissional. **Ciências e Cognição/Science and Cognition**, v. 8, 2011.

KENSKI, V. M. Aprendizagem mediada pela tecnologia. **Revista diálogo educacional**, Curitiba, v. 4, n. 10, p. 47-56, 2003.

_____. **Educação e Tecnologias: o novo ritmo da informação**. 8 ed. Campinas, SP: Papiros 2012.

MARCONI, M. d. A.; LAKATOS, E. M. **Fundamentos de metodologia de científica**. 5 ed. São Paulo: Atlas, 2003.

MARIANI, V. C.; PRETO, T. M.; GUEDES, A. L. P. Utilização do Maple, MATLAB e SCILAB nos Cursos de Engenharia. **In**: XXXIII COBENGE 2005 – Congresso Brasileiro de Ensino de Engenharia, 2005.

MARTINS, K. L. Teorias de aprendizagem e avaliação de software educativo. (Monografia) Especialização em Informática Educativa – Universidade Federal do Ceará, 2002.

MOTA, R. P. B. Código livre Scilab para o ensino de Cálculo Numérico. In: XXII SBIE - Simpósio Brasileiro de Informática na Educação e XVII WIE - Workshop de Informática na Escola. Aracaju, p.600-609. 2011.

OLIVEIRA, C. C.; COSTA, J. W.; MOREIRA, M. **Ambientes informatizados de aprendizagem**: produção e avaliação de software educativo. Papirus, 2001. Disponível em: < <https://books.google.com.br/books?id=TNPzAAAACAAJ> >. Acesso em: 2 de out. de 2015.

OLIVEIRA, H.; DOMINGOS, A. Software no ensino e aprendizagem da Matemática: Algumas ideias para discussão. **Tecnologias e Educação Matemática**, p. 279-285. 2008.

PAPERT, S. **Mindstorms**: Children, computers, and powerful ideas. Basic Books, Inc., 1980.

PENTEADO, M. et al. **A informática em ação**: formação de professores, pesquisa e extensão. Olho d'Água, 2000.

PENTEADO, M. G.; BORBA, M. C.; GRACIAS, T. S. Informática como veículo para mudança. **Rev. Zetetiké**, CEMPEM-UNICAMP, v. 6, n. 10, p. 77-84, 1998.

RAMOS, E. M.; MENDONÇA, I. J. **O fundamental na avaliação da qualidade do software educacional**. Laboratório de Software Educacional - EDUGRAF, 1991.

RIBEIRO, M. J. B.; PONTE, J. P. A formação em novas tecnologias e as concepções e práticas dos professores de Matemática. **Quadrante**, v. 2, p. 3 - 26, 2000.

ROCHA, A. R.; CAMPOS, G. H. B. d. Avaliação da qualidade de software educacional. **Aberto**, v. 12, p. 32-44, 1993.

RODRIGUES LONGO, G. A.; LAMBLÉM, R. L.; DUARTE, M. A. Q. Uso do Scilab e do Octave como opções para o ensino da matemática e resolução de problemas em análise numérica. **Anais do Encontro de Iniciação Científica - ENIC**, v. 1, n. 4, 2012.

SANT'ANA, C. d. C.; AMARAL, R. B.; BORBA, M. d. C. O uso de softwares na prática profissional do professor de matemática. **Ciência & Educação**, v. 18, n. 3, p. 527 - 542, 2012.

SILVA, E. M.; CUNHA, J. P. V. S. Scilab, scicos e rtool: Softwares livres no ensino de engenharia elétrica. In: Anais do Congresso Brasileiro de Automática, p.1620 - 1625. 2006.

SILVA, S. R. X. O uso do scilab como ferramenta para o ensino de cálculo numérico
In: XLI COBENGE - Congresso Brasileiro de Ensino em Engenharia. Gramado, RS.
2013.

SOUZA, J. A. d.; DANDOLINI, G. A. Utilizando simulação computacional como
estratégia de ensino: estudo de caso. **RENOTE**, v. 7, n. 1, 2009.

TIKHOMIROV, O. K. The psychological consequences of computerization. In:
WERTSCH, J. V. (Org.). **The concept of activity in Soviet psychology**. New York:
M. E. Shaper: Inc, p.256-278, 1981.

TRINDADE, J. M. Scilab, Geogebra e Winplot como recurso pedagógico no ensino
de matrizes, determinantes e geometria analítica. Dissertação (Mestrado) –
Universidade Federal do Maranhão. 2013.

TSUKUMO, A. N. et al. Qualidade de software: visões de produto e processo de
software. **In:** VII CITS - Conferência Internacional de Tecnologia de Software.
Curitiba, Paraná, 1997.

VALENTE, J. A. Diferentes usos do computador na educação. In: (Org.).
Computadores e Conhecimento: repensando a educação. Campinas:
Unicamp/Nied, p.1-23, 1993.

_____. **Informática na educação: instrucionismo x construcionismo**. Manuscrito
não publicado. NIED: UNICAMP, 1997a.

_____. O uso inteligente do computador na educação. **Revista Pátio**, v. 1, p. 19-
21, 1997b.

_____. **O computador na sociedade do conhecimento**: UNICAMP/NIED,
Campinas, 1999.

_____. Pesquisa, comunicação e aprendizagem com o computador. **O papel**, 2005.

VALENTE, J. A.; ALMEIDA, F. J. Visão analítica da informática na educação no
Brasil. **Revista Brasileira de Informática na Educação**, v. 1, p. 45-60, 1997.

VIEIRA, F. M. S. **Avaliação de software educativo: reflexões para uma análise
criteriosa**. Proinfo, 1999.

ANEXO A – Lista de comandos do Scilab

Tabela A.1 - Lista de comandos do Scilab

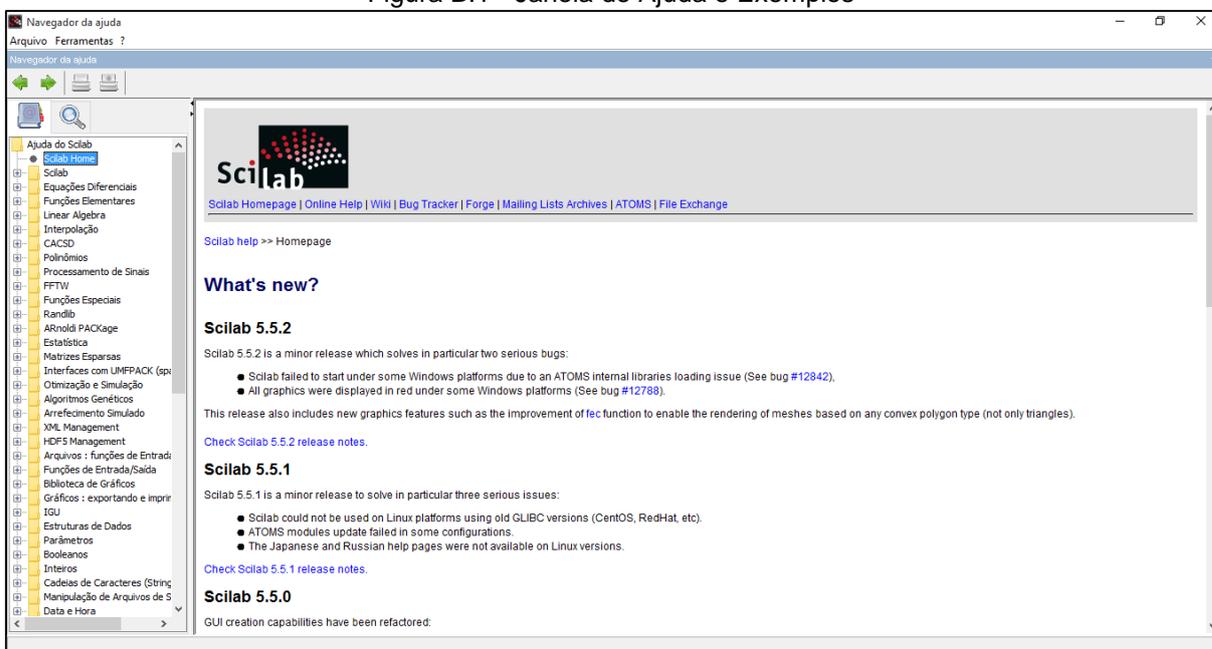
COMANDOS	DESCRIÇÃO
<code>abs(x)</code>	Módulo (argumento) de x
<code>acos(x)</code>	Arco – cosseno de x , em radianos
<code>acosh(x)</code>	Arco – cosseno hiperbólico de x , em radianos
<code>asin(x)</code>	Arco – seno de x , em radiano
<code>asinh(x)</code>	Arco – seno hiperbólico de x , em radiano
<code>atan(x)</code>	Arco – tangente de x , em radiano
<code>atanh(x)</code>	Arco – tangente hiperbólico de x , em radianos
<code>binomial(p,n)</code>	Binomial normalizado de p classe n^*
<code>ceil(x)</code>	Arredondamento para o maior inteiro posterior
<code>cos(x)</code>	Cosseno de x , com x em radiano
<code>cosh(x)</code>	Cosseno hiperbólico de x , com x em radianos
<code>cotg(x)</code>	Cotangente de x , com x em radianos
<code>coth(x)</code>	Cotangente hiperbólica de x , com x em radianos
<code>cumprod(v)</code>	Produto cumulativo do vetor v
<code>cumsum(v)</code>	Soma cumulativa do vetor v
<code>diff(v)</code>	Diferencial discreta do vetor v
<code>eval(s)</code>	Valor numérico de uma expressão de strings*
<code>exp(x)</code>	Exponencial euleriana de x
<code>factor(N)</code>	Fatores primos de N
<code>find(k 's' V)</code>	Encontra os índices dos elementos de V que tornam a comparação verdadeira ('s' \rightarrow ==, <, >, ~=, ...)
<code>fix(x)</code>	Encontra os índices dos elementos de V que tornam a comparação verdadeira
<code>floor(x)</code>	Arredondamento para o maior inteiro anterior
<code>gsort(V,'r/c','i/d')</code>	Ordenação dos elementos de V
<code>imag(z)</code>	Coeficiente da parte imaginária de z
<code>int(N)</code>	Parte inteira de N
<code>linspace(A,B,N)</code>	Vetor com N pontos entre A e B inclusos
<code>log(x)</code>	Logaritmo neperiano de x
<code>log10(x)</code>	Logaritmo decimal de x
<code>log2(x)</code>	Logaritmo na base 2 de x

logspace (A,B,N)	Vetor de N pontos com espaço logarítmico entre A e B incluso
max(v)	Máximo valor contido no vetor v
min(v)	Mínimo valor contido no vetor v
norm(A)	Norma de A
perms(v)	Todas as permutações do vetor v
primes(N)	Todos os números primos até N
prod(v)	Produto de todos os elementos de v
real(z)	Parte real do número complexo
round(N)	Número inteiro mais próximo de N
sin(x)	Seno de x, x em radianos
sinh(x)	Seno hiperbólico de x, x em radiano
sqrt(x)	Raiz quadrada de x
tan(x)	Tangente de x, x em radianos
tanh(x)	Tangente hiperbólica de x, x em radianos

Fonte: Elaborado pelo autor.

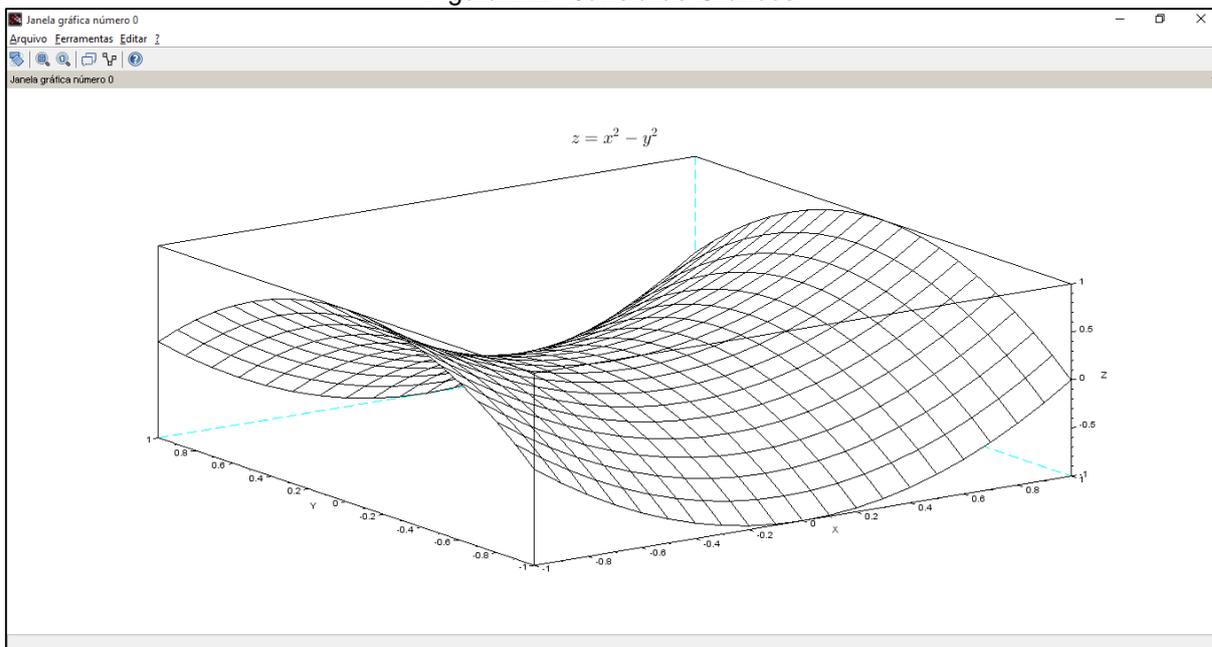
ANEXO B – Janelas de recursos do Scilab

Figura B.1 - Janela de Ajuda e Exemplos



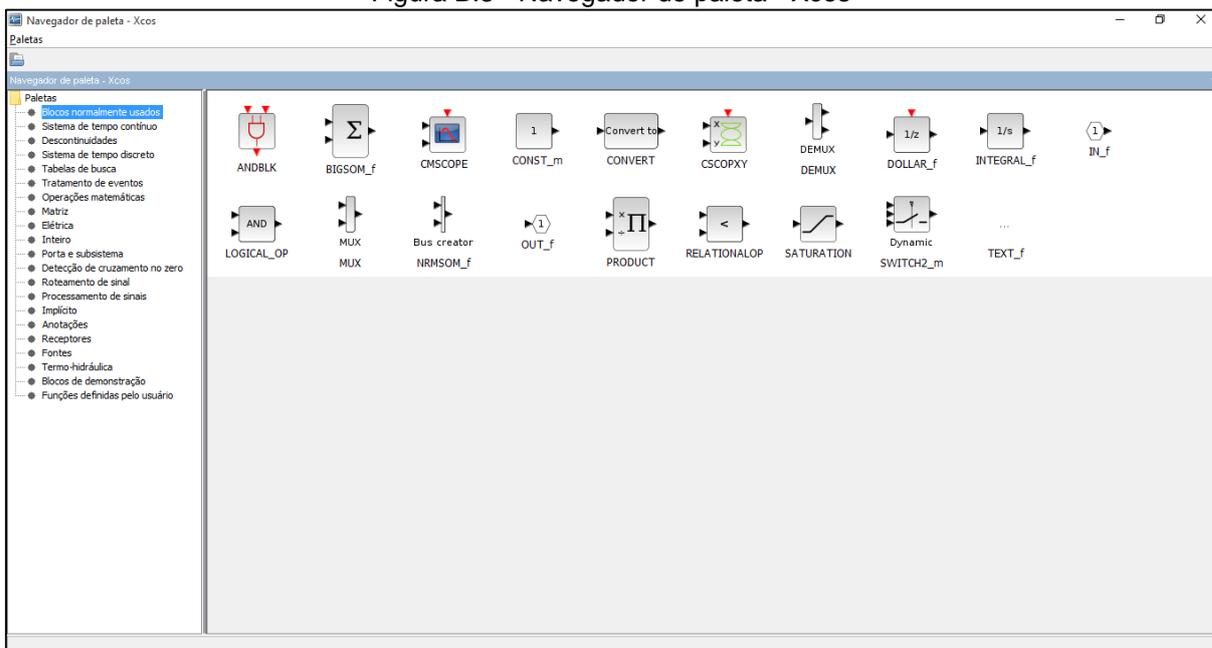
Fonte: Elaborado pelo autor.

Figura B.2 - Janela de Gráficos



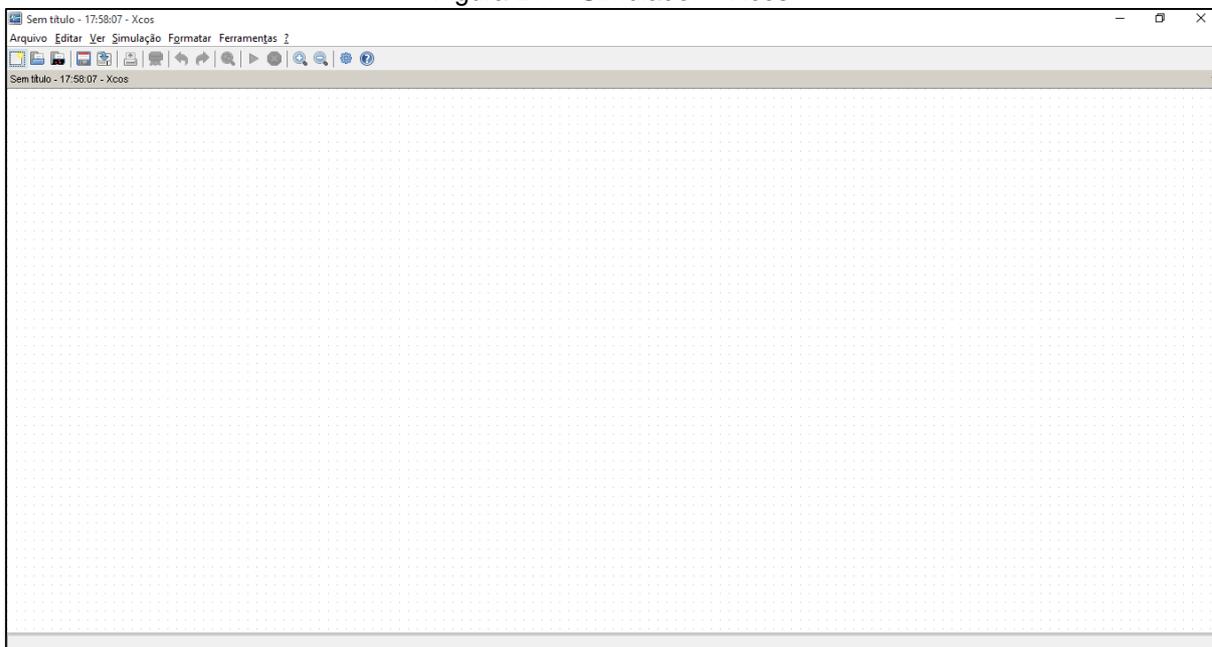
Fonte: Elaborado pelo autor.

Figura B.3 - Navegador de paleta - Xcos



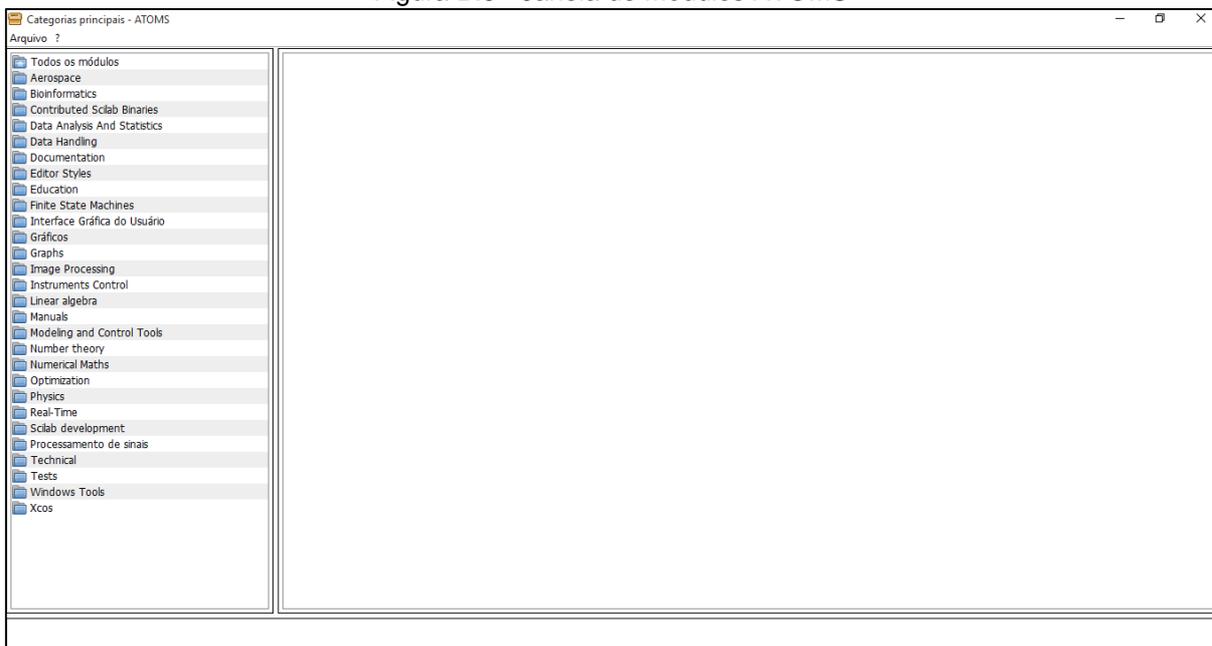
Fonte: Elaborado pelo autor.

Figura B.4 - Simulador – Xcos



Fonte: Elaborado pelo autor.

Figura B.5 - Janela de Módulos ATOMS



Fonte: Elaborado pelo autor.