



GUILHERME DOS REIS BAVIERA

**ANÁLISE DE ESFORÇO PARA NAVEGAÇÃO
EM CONTEÚDO MATEMÁTICO NA WEB COM
O PLUGIN ACCESS8MATH**

LAVRAS – MG

2020

GUILHERME DOS REIS BAVIERA

**ANÁLISE DE ESFORÇO PARA NAVEGAÇÃO EM CONTEÚDO
MATEMÁTICO NA WEB COM O PLUGIN ACCESS8MATH**

Monografia apresentada à Universidade Federal de
Lavras, como parte das exigências do Curso de
Sistemas de Informação, para a obtenção do título
de Bacharel.

Prof. Dr. André Pimenta Freire
Orientador

Hérion Manollo Cândido Guedes
Coorientador

LAVRAS – MG

2020

GUILHERME DOS REIS BAVIERA

**ANÁLISE DE ESFORÇO PARA NAVEGAÇÃO EM CONTEÚDO
MATEMÁTICO NA WEB COM O PLUGIN ACCESS8MATH**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do Curso de Sistemas de Informação, para a obtenção do título de Bacharel.

APROVADA em 14 de Agosto de 2020.

Prof. Dr. André Pimenta Freire	UFLA
Prof. Dra. Paula Christina Figueira Cardoso	UFLA
BSc. Jorge Sasaki Resende Ssilva	UFLA



Prof. Dr. André Pimenta Freire
Orientador

Hérilon Manollo Cândido Guedes
Co-Orientador

**LAVRAS – MG
2020**

Dedico este trabalho à minha família.

AGRADECIMENTOS

Agradeço a minha família por todo o apoio concedido durante toda a graduação. Aos professores que dedicaram seu tempo a passar o conhecimento necessário nessa trajetória. Agradecimentos especiais ao professor André Pimenta Freire e ao coorientador do trabalho Hérlon Manollo Cândido Guedes, pela paciência, atenção e dedicação empenhadas na realização deste trabalho.

RESUMO

Ensinar matemática para pessoas com deficiência visual possui algumas limitações, dentre elas a leitura e entendimento de fórmulas matemáticas. Recursos de Tecnologia Assistiva como leitores de tela podem auxiliar a melhorar o acesso a pessoas com deficiência visual. Entretanto, esses sistemas ainda têm limitações para fazer a leitura, uma vez que a síntese por conteúdo em áudio pode causar sobrecarga de memória aos usuários e gasto maior de tempo para executar as tarefas. O presente estudo visa identificar o esforço necessário para que usuários com deficiência visual utilizem o leitor de telas NVDA com o plugin para leitura de conteúdo matemático Access8Math. As estimativas foram realizadas utilizando o modelo GOMS (Goals, Objectives, Methods, Selection Rules) com KLM (Keystroke Level Method). Foram feitas análises com estudos anteriores utilizando os leitores NVDA, JAWS e ChromeVox, e comparações com o plugin Access8Math e uma versão implementada no grupo de pesquisa com recursos de navegação contextual. Os resultados mostraram que o uso de navegação contextual, com leitura baseada na visão geral e aprofundamento por escolha do usuário, pode ter tempo comparável a formas de navegação que utilizam teclas de atalho para elementos matemáticos específicos, como raízes e somatórios. Os resultados obtidos são importantes para guiar os testes com usuários que serão realizados em próximas etapas da pesquisa.

Palavras-chave: Matemática; Deficiência Visual; Leitor de telas.

ABSTRACT

Teaching mathematics to people with visual impairments still has some limitations, including reading and understanding mathematical formulas. Assistive technology features like screen readers can help improve access for people with visual impairments. However, these systems still have limitations for reading, since audio synthesis of content can cause memory overload for users and more time to perform to perform tasks. This study aimed to identify the effort required for visually impaired users to use the NVDA screen reader with the Access8Math mathematical content reading plugin. The estimated times were made using the GOMS model (Goals, Objectives, Methods, Selection Rules) with KLM (Keystroke Level Method). Analyses were made comparing results from previous studies using NVDA, JAWS and ChromeVox readers, relating with the Access8Math plugin and a version implemented in the research group with contextual navigation features. The results showed that the use of contextual navigation, with reading based on the overview and deepening by choice of the user, can have time comparable to forms of navigation that use shortcut keys for specific mathematical elements, such as roots and summation. The results obtained are important to guide the tests with users that will be carried out in the next stages of the research.

Keywords: Mathematics. Visually Impaired. Screen reader.

LISTA DE FIGURAS

Figura 2.1 – Exemplo de notação matemática em MathML	23
Figura 2.2 – Representação de uma notação matemática no modelo linear .	24
Figura 2.3 – Representação de uma notação matemática no modelo nave- gação em árvore	25
Figura 2.4 – Representação de uma notação matemática no modelo contex- tual	26
Figura 3.1 – Menu de configurações do Access8Math	35
Figura 3.2 – Arquivo math.rule	37
Figura 3.3 – Arquivo unicode.dic	37
Figura 3.4 – Esboço da página HTML criada para os testes	39
Figura 4.1 – Menu de interação para conteúdo matemático	41
Figura 4.2 – Modelo Leitura original - GOMS para a tarefa 1 usando o Access8Math	45
Figura 4.3 – Modelo Leitura original - GOMS para a tarefa 2 usando o Access8Math	45
Figura 4.4 – Modelo Leitura original - GOMS para a tarefa 3 usando o Access8Math	46
Figura 4.5 – Modelo Evitar leitura repetida - GOMS para a tarefa 1 usando o Access8Math	46
Figura 4.6 – Modelo Evitar leitura repetida - GOMS para a tarefa 2 usando o Access8Math	47
Figura 4.7 – Modelo Evitar leitura repetida - GOMS para a tarefa 3 usando o Access8Math	47
Figura 4.8 – Modelo Navegação contextual - GOMS para a tarefa 1 usando a nova versão Access8Math	48
Figura 4.9 – Modelo Navegação contextual - GOMS para a tarefa 2 usando a nova versão Access8Math	49

Figura 4.10 – Modelo Navegação contextual - GOMS para a tarefa 3 usando
a nova versão Access8Math 49

LISTA DE TABELAS

Tabela 2.1 – Tempos de estimativa de esforço	30
Tabela 3.1 – Tarefas	38
Tabela 3.2 – Variáveis KLM	38
Tabela 4.1 – Comandos na versão original do Access8Math	42
Tabela 4.2 – Comandos na versão disponibilizada por Guedes (2019) . . .	43
Tabela 4.3 – Tempos de estimativa de esforço de leitores de telas anteriores e os 3 modelos utilizando o Access8Math	52

SUMÁRIO

1	INTRODUÇÃO	17
2	REFERENCIAL TEÓRICO	21
2.1	Acessibilidade na web e leitura de formulas matemáticas	21
2.2	Modelagem de Tarefas com GOMS e KLM	27
2.3	Trabalhos relacionados	29
3	METODOLOGIA	33
3.1	Desenho do estudo	33
3.2	Análise do plugin Access8Math	34
3.3	Tradução da leitura e interface do Access8Math	36
3.4	Análise de tarefas e estimativa de esforço	37
4	RESULTADOS E DISCUSSÃO	41
4.1	Análise do Access8Math	41
4.2	Expectativa de esforço para navegação no Access8Math	43
4.3	Comparação da expectativa de esforço do Access8Math com outros leitores de tela	47
5	CONCLUSÃO	55
	REFERÊNCIAS	57

1 INTRODUÇÃO

As práticas de ensino têm incorporado cada vez mais a utilização de dispositivos tecnológicos para facilitar e melhorar a experiência do processo de ensino-aprendizagem, permitindo, assim, novas técnicas e metodologias para efetuar o mesmo. Entretanto, na educação há desafios a serem superados para prover acessibilidade de conteúdos para pessoas com deficiência. Em particular, diferentes áreas do conhecimento demandam diferentes formas de serem abordadas, de forma a organizar o raciocínio e o aprendizado. Na Matemática, por exemplo, é preciso abordar de maneira eficaz a utilização de forma acessível de seus símbolos e notações.

Tornar o conteúdo matemático disponível e acessível a todos ainda não é uma tarefa simples. Mesmo com os avanços nas pesquisas de Artefatos de Tecnologia Assistiva, a acessibilidade na matemática ainda apresenta algumas dificuldades. Uma dessas dificuldades é quanto à formatação das fórmulas matemáticas. De acordo com as *guidelines* do W3C (W3C Web Accessibility Initiative, 2008), a linguagem de marcação recomendada para a criação e manipulação de fórmulas matemáticas é a MathML (MINER; ION; CARLISLE, 2014).

Entretanto, nem todos os materiais envolvendo fórmulas matemáticas são disponibilizados seguindo essa recomendação. Além disso, outro problema é a falta de recursos satisfatórios para efetuar a leitura matemática. Essa escassez de ferramentas eficazes para a leitura matemática pode comprometer o aprendizado e formação do indivíduo nessa área de conhecimento.

Com o intuito de facilitar o desenvolvimento do aprendizado das pessoas são utilizados diversos sistemas interativos, tendo em vista a facilidade de acesso e uso dos mesmos. No que tange às pessoas com deficiência visual há destaque para os recursos de Tecnologia Assistiva em tarefas que demandam o uso da informação. Dentre os recursos destacam-se os leitores de telas, que têm como objetivo sintetizar o formato de texto em áudio voz.

Ainda que alguns estudos tenham sido realizados abrangendo o desenvolvimento e análise de Artefatos de Tecnologia Assistiva, especialmente leitores de telas, pouco ainda há esclarecido a respeito do esforço demandado por partes dos usuários para realizarem tarefas cotidianas, dentre elas a leitura de fórmulas matemáticas.

A utilização de leitores de telas pode ser um importante motivador quanto ao aprendizado e autonomia de usuários com deficiência visual, proporcionando maior independência e integração social dos mesmos ao acessar conteúdos digitais. Existem diversos leitores de telas disponíveis no mercado, dentre eles destacam-se: JAWS (*Job Access With Speech*)¹, ChromeVox² e o NVDA (*NonVisual Desktop Access*)³ Estes oferecem suporte a leitura de conteúdo matemático na Web.

Segundo pesquisa realizada pela empresa de consultoria em tecnologia Everis Brasil (EVERIS BRASIL, 2019), o NVDA é o leitor de telas mais utilizado no Brasil. Este leitor permite que os usuários naveguem por itens, por cabeçalhos ou por links da página, possibilitando comandos diversos para o usuário conduzir a navegação. O NVDA permite integração de *plugins* em sua interface.

Um *plugin* externo que o NVDA permite instalação é o MathPlayer⁴. Em versões anteriores, a leitura de conteúdo matemático era feita de forma contínua, sem permitir ao usuário controle sobre o conteúdo que está sendo lido. Em versões mais recentes, o *plugin* permite a leitura de partes de fórmulas e detalhamento em partes específicas.

Outro *plugin* do NVDA é o Access8Math⁵ que permite leitura de conteúdo matemático estruturado em MathML na WEB. Ao acessar um determinado conteúdo matemático o *plugin* permite a abertura de uma janela de interação onde o usuário pode controlar a navegação sobre o conteúdo matemático e até mesmo

¹ Disponível em <<https://www.freedomscientific.com/Products/software/JAWS/>>.

² Disponível em <<https://www.chromevox.com/>>.

³ Disponível em <<https://www.nvaccess.org/>>.

⁴ Disponível em <<http://www.dessci.com/en/products/mathplayer/>>.

⁵ Disponível em <<https://addons.nvda-project.org/addons/access8math.en.html>>

copiá-lo. Caso o Access8Math esteja instalado em conjunto com o MathPlayer é possível realizar leitura no editor de textos Word.

Com o intuito de avaliar e comparar o esforço necessário para que os usuários utilizassem os leitores de telas para o conteúdo matemático, um estudo anterior do Núcleo de pesquisa ALCANCE realizou a análise dos leitores de tela JAWS, NVDA e ChromeVox (SILVA et al., 2017), seguido de outro trabalho com a análise do impacto do tempo com recursos de navegação melhorados em uma nova versão do ChromeVox, a versão NavMatBR(SILVA et al., 2018). Contudo, os estudos não contemplaram o plugin Access8Math, que é um recurso de código aberto que funciona com o NVDA.

O modelo que possibilitou analisar os leitores anteriores é denominado modelo de tarefas GOMS-KLM, que traz em seu contexto métricas de tempos que permitem avaliar e comparar o desempenho para a realização de atividades em diferentes fórmulas matemáticas, utilizando os recursos que cada navegador dispõe.

O Access8Math dispõe de uma versão original disponibilizada pelo seu desenvolvedor e outra em desenvolvimento em conjunto com este trabalho, ambas possuem distintas abordagens de navegação em conteúdo matemático Web.

Para realizar as tarefas utilizado o mesmo modelo que os estudos anteriores, possibilitará, não só avaliar e mensurar o tempo gasto pelo *plugin*, como também comparar com os demais resultados obtidos com os trabalhos anteriores.

A combinação da análise dos resultados do modelo GOMS-KLM juntamente com os recursos disponíveis para navegação disponibilizadas pelo *plugin* permitirá também uma visão sobre lacunas a serem consideradas como pontos de melhorias.

Sendo assim, o objetivo deste trabalho de conclusão de curso foi de estender os trabalhos anteriores e realizar uma análise dos recursos de navegação em fórmulas matemáticas do plugin Access8Math para o NVDA, e comparar as esti-

mativas de esforço de navegação em fórmulas matemáticas deste estudo com já realizados utilizando o modelo de tarefas GOMS-KLM (CARD; MORAN; NEWELL, 1980).

Os resultados obtidos neste trabalho com a análise do *plugin* Access8Math indicam que há possibilidade de redução, tanto em tempo, quanto em esforço por parte do usuário para a execução de tarefas relacionadas a leitura de conteúdo matemático na Web.

O trabalho está organizado da seguinte forma: Na seção 2 são apresentados conceitos relacionados com o trabalho. Na seção 3 a metodologia utilizada no trabalho é apresentada. Na seção 4 os resultados alcançados são apresentados e, para terminar, a conclusão é discutida na seção 5.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta os principais conceitos utilizados neste trabalho, abordando Acessibilidade na Web e leitura de fórmulas matemáticas, Modelagem de tarefas com GOMS e KLM e trabalhos relacionados.

2.1 Acessibilidade na web e leitura de formulas matemáticas

A tecnologia é utilizada como um facilitador para as atividades do dia a dia em diversos contextos é importante para as pessoas. Porém, de acordo com RADABAUCH (1993), no caso de pessoas com deficiência, a tecnologia não só age como um agente facilitador, mas torna as coisas possíveis de serem realizadas.

De modo específico, por se tratar de uma linguagem que se apoia fortemente na comunicação escrita visual, conforme Stevens, Edwards e Harling (1997), a matemática é uma área repleta de desafios relacionados à acessibilidade, especialmente para pessoas com deficiência visual.

Neste contexto, surgem os recursos de Tecnologia Assistiva (TA) para suavizar as barreiras presentes no cotidiano das pessoas com deficiência. As consequências da deficiência são um fato de construção social, resultante de barreiras presentes no ambiente, causando impactos de diversos níveis na vida dos indivíduos, segundo Cook e Polgar (2014).

Dentre os diversos recursos de TA desenvolvidos destacam-se os leitores de tela. Esse recurso é utilizado por muitas pessoas com deficiência visual ou baixa visão. Essas ferramentas se caracterizam por transmitir, através de um sintetizador de voz, o conteúdo presente em uma tela.

A interação dos usuários com os leitores de tela ocorre, geralmente, através do teclado como forma de entrada. A partir dos comandos enviados, o leitor de telas é capaz de realizar as ações solicitadas pelo usuário e transmitir o efeito dessas ações através da leitura da tela. Habitualmente, essa leitura se dá de forma

sequencial, tendo início na parte superior a esquerda seguindo para a direita e para baixo.

As formas de navegação mais comumente ofertadas pelos leitores de tela são através de links, cabeçalhos, parágrafos, dentre outras. Essas formas de navegação são utilizadas para diminuir o tempo que seria necessário para ler todo o conteúdo de forma sequencial, que pode ser cansativo para os usuários, que muitas vezes estão interessados em somente parte do conteúdo.

Entretanto, para que ocorra tais navegações, o conteúdo a ser lido precisa estar estruturado conforme as recomendações de acessibilidade, tais como do WCAG (*Web Content Accessibility Guidelines*) (W3C Web Accessibility Initiative, 2008).

As páginas Web são desenvolvidas com marcações HTML (*HyperText Markup Language*) (FAULKNER et al., 2017). Essas marcações definem toda a estrutura da página, como links, parágrafos, conteúdo multimídia, formulários, dentre outros. O MathML (*Mathematical Markup Language*) (MINER; ION; CARLISLE, 2014) permite criar conteúdos de notações matemáticas a partir de tags de marcação de uma forma estruturada, semelhante às estruturas feitas em HTML para as páginas Web.

A Figura 2.1 demonstra a representação da Equação 2.1 em MathML. Observa-se o uso de marcação com *tags* que simbolizam cada elemento de uma notação matemática, com operadores, símbolos e números.

$$x^2 + 4x + 4 = 0 \quad (2.1)$$

Há diversos leitores de telas no mercado, livres e comerciais. Alguns dos principais leitores são: JAWS (*Job Access With Speech*), NVDA (*NonVisual Desk-*

Figura 2.1 – Exemplo de notação matemática em MathML

```

1 <mrow>
2   <mrow>
3     <msup>
4       <mi> x </mi>
5       <mn> 2 </mn>
6     </msup>
7     <mo> + </mo>
8     <mrow>
9       <mn> 4 </mn>
10      <mo> & InvisibleTimes; </mo>
11      <mi> x </mi>
12    </mrow>
13    <mo> + </mo>
14    <mn> 4 </mn>
15  </mrow>
16  <mo> = </mo>
17  <mn> 0 </mn>
18 </mrow>

```

Fonte: <https://www.w3.org/Math/whatIsMathML.html>

top Access), DosVox¹, Virtual Vision², Voice Over, Orca³, ChromeVox⁴, dentre outros. Nem todos os leitores de tela possuem recursos para leitura de conteúdo matemático. Outros ainda são limitados à determinados sistemas operacionais ou navegadores web.

Quanto à leitura matemática, a literatura atualmente aborda a utilização de basicamente três formas distintas de navegação: linear, contextual e em árvore.

No modo linear, a leitura é feita de forma progressiva, linear, direta e ininterrupta sem a possibilidade de navegação ou exploração de elementos individuais. Essa abordagem pode ser inapropriada para conteúdos matemáticos para pessoas cegas, que podem ser complexos e extensos, envolvendo vários elementos em uma mesma notação, com dificuldade para memorização sem apoio visual. Segundo

¹ Disponível em <<http://intervox.nce.ufrj.br/dosvox/>>.

² Disponível em <<https://www.virtualvision.com.br/>>.

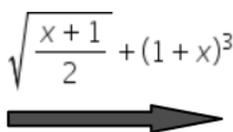
³ Disponível em <<https://wiki.gnome.org/Projects/Orca>>.

⁴ Disponível em <<https://www.chromevox.com/>>.

Sorge et al. (2014) esta abordagem é incapaz de prover ao usuário a possibilidade de entender e interagir com o conteúdo.

Na Figura 2.2 temos a representação de uma leitura linear da expressão matemática $\sqrt{\frac{x+1}{2}} + (1+x)^3$.

Figura 2.2 – Representação de uma notação matemática no modelo linear


$$\sqrt{\frac{x+1}{2}} + (1+x)^3$$

Fonte: do autor 2020.

A estratégia de navegação em árvore teve como trabalho precursor o estudo de Raman (1994), com um leitor de telas denominado ASTER (*Audio System for Technical Readings*). Outro trabalho relacionado a esta abordagem é o de Gaura (2002), um sistema especializado denominado REMathEx (*Reader and Editor of the Mathematical Expressions*).

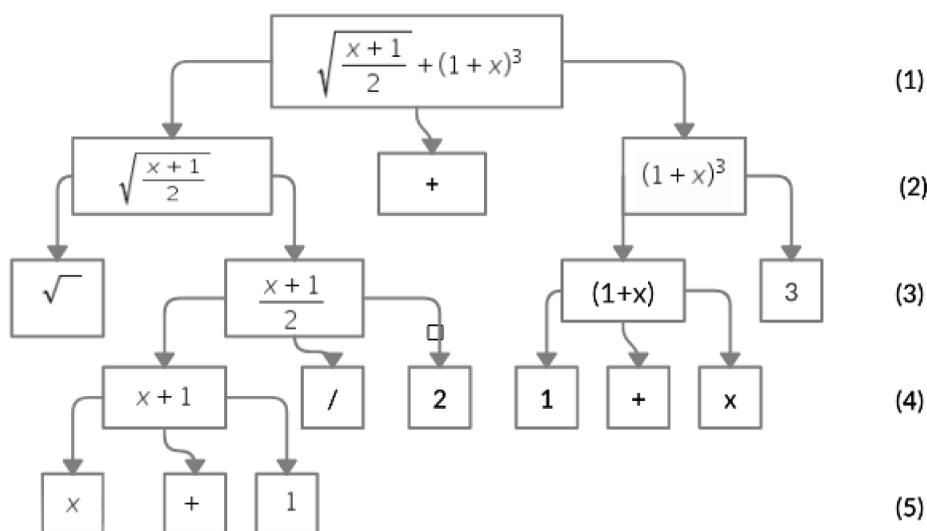
Ressalta-se que o JAWS, um dos leitores mais difundidos no mercado também utiliza esta abordagem de navegação. Tal técnica permite a navegação entre elementos seguindo a ideia da estruturação em árvore, um tipo de estrutura bem difundido no meio computacional. Alguns comandos de navegação estão mencionados a seguir:

- Marcar o nó atual;
- Ir até o nó marcado.
- Ir para o elemento pai;
- Navegar até o elemento filho mais a esquerda;
- Navegar até o elemento filho mais a direita;
- Ir para o próximo elemento irmão;

- Voltar ao elemento irmão anterior.

Na Figura 2.3 temos a representação de uma estrutura em árvore da expressão matemática $\sqrt{\frac{x+1}{2}} + (1+x)^3$.

Figura 2.3 – Representação de uma notação matemática no modelo navegação em árvore



Fonte: do autor 2020.

De acordo com Stevens, Edwards e Harling (1997) e Sorge et al. (2014), para que seja efetivo a utilização da abordagem de navegação em árvore há a necessidade de entendimento completo do funcionamento desta estrutura. Já Gaura (2002) discorda desta abordagem, salientando que a navegação não necessita de conhecimento prévio sobre a estrutura para que a mesma seja eficiente no que tange a conteúdos matemáticos.

Por fim, Stevens, Edwards e Harling (1997) apresenta a navegação contextual. Neste trabalho os autores apresentam o MathTalk, um leitor baseado na abstração de termos complexos, separando a fórmula matemática em elementos simples e complexos. Em um primeiro instante, o conteúdo é lido de forma linear, apresentando cada elemento simples, e os complexos são substituídos por significados representativos, como por exemplo, "x é igual a uma função", como forma

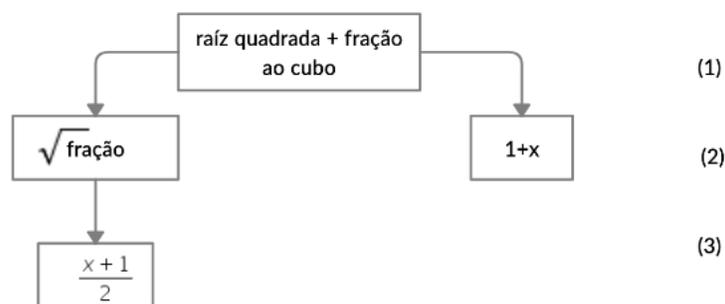
de sintetizar conteúdos que seriam longos, dando ao usuário a opção de ter uma compreensão mais global da fórmula, e então ler as partes de forma detalhada.

Alguns comandos disponíveis para possibilitar a navegação do usuário são: *speak* (s), *next* (n), *previous* (p), *into* (i), *out-of* (o) e *expression* (e). Estes comandos podem ser combinados entre si, formando assim uma gama ainda maior de possibilidades de navegação nos conteúdos matemáticos. Desta maneira reduz-se possível ambiguidade e dá maior poder de ação ao usuário, aumentando a experiência de usabilidade com o leitor MathTalk.

Stevens, Edwards e Harling (1997) ressalta que a abordagem contextual pode trazer benefícios como redução de informações simultâneas, de carga mental e ambiguidade. A redução neste aspecto caracteriza-se pela leitura do conteúdo da fórmula ser feita em pequenos trechos.

Na Figura 2.4 temos a representação do modelo contextual da expressão matemática $\sqrt{\frac{x+1}{2}} + (1+x)^3$.

Figura 2.4 – Representação de uma notação matemática no modelo contextual



Fonte: do autor 2020.

Ainda há uma lacuna sobre qual o modelo ideal para aplicar aos leitores de telas de forma a evidenciar uma real melhoria na eficácia da utilização de tais ferramentas assistivas. Existem problemas de limitações relacionados a idiomas.

Cada idioma tem suas particularidades culturais sobre como fazer a leitura de conteúdos matemáticos, impactando a forma como é lido em cada um deles (WONGKIA; NARUEDOMKUL; CERCONE, 2009). Tomemos um simples

exemplo na notação x^2 , a mesma pode ser lida como "xis elevado ao quadrado", "xis ao quadrado" ou até mesmo "xis elevado a dois".

2.2 Modelagem de Tarefas com GOMS e KLM

A modelagem de tarefas consiste na descrição passo a passo de tarefas que devem ser executadas para que o usuário atinja algum objetivo, compondo um modelo que descreve as ações, condições para execução e regras para a realização das tarefas em direção ao objetivo. Estes modelos podem ser utilizados para criar interfaces e avaliar a interatividade de aplicativos (CARD; MORAN; NEWELL, 1980).

Uma técnica de modelagem de tarefas é a abordagem GOMS (*Goals, Operators, Methods and Selection Rules*). Essa abordagem está pautada em quatro pilares, são eles: Metas (*Goals*) aquilo que se deseja que o usuário faça, Operadores (*Operators*) que são ações humanas básicas que o usuário realiza como exemplo pressionar uma tecla ou escutar algo, Métodos (*Methods*) que são a sequência de passos para atingir uma meta e por fim as Regras de seleção (*Selection Rules*), que definem as regras de seleção do método a ser utilizado (CARD; NEWELL; MORAN, 1983).

Os objetivos dos modelos GOMS são definidos ao início de cada tarefa, descrevendo de forma direta qual a finalidade da mesma. As metas (*Goals*) dos modelos são definidas em cada ação dentro do modelo, como por exemplo, encontrar a expressão matemática desejada na tela, ativar o modo de interação do leitor ou navegar na expressão para encontrar determinado elemento matemático.

O operador (*Operators*) utilizado para realizar cada tarefa foi o pressionamento de teclas que produzam efeito de navegação no *plugin* através do teclado de entrada do computador. Os métodos (*Methods*) definem o passo a passo de pressionamento de teclas para realizar a navegação. Não foram utilizadas regras de seleção (*Selection Rules*).

Há um modelo derivado do GOMS denominado *Keystroke-Level Model* (KLM). KLM usa as teclas pressionadas, movimentos do mouse ou teclas do mouse como meio de analisar e avaliar uma tarefa. Ao utilizar este modelo é possível somar a estimativa de tempo de forma quantitativa. (CARD; MORAN; NEWELL, 1980)

Definiu-se três operadores para realizar a análise quantitativa do modelo: o primeiro operador é o de pressionamento de teclas, definido com a letra K. O segundo operador tem relação com a espera mental dos usuários, definido com a letra M. O terceiro e último operador definido é sobre o tempo do leitor de telas ler cada palavra, definido como W. Esses operadores são baseados no trabalho de Card, Moran e Newell (1980).

Para exemplificar um modelo, tomemos a notação 2.2 onde pode-se definir o objetivo encontrar o expoente de cada base. As metas seriam navegar na tela até encontrar a expressão desejada, ativar o modo de leitura desejado e navegar na expressão até que o usuário possa atingir o objetivo. Durante a execução das metas os operadores do modelo são incrementados de acordo com cada pressionamento de tecla, cada leitura de palavras por parte do sintetizador de voz do *plugin* e espera mental por parte do usuário.

$$x = a^2 + b^2 \quad (2.2)$$

Utilizando do modelo GOMS-KLM, este trabalho teve como foco a descrição e uma estimativa de tempo das tarefas a serem realizadas na navegação em conteúdo matemático com o *plugin* Access8Math, utilizando dois modos de navegação, um que considera a navegação utilizando a versão original do mesmo, disponibilizada pelo desenvolvedor, e um segundo modo de navegação disponibilizada em uma versão atualizada, disponibilizada pelo trabalho de Guedes (2019).

2.3 Trabalhos relacionados

A leitura contínua e linear de fórmulas matemáticas por áudio pode apresentar muitas dificuldades para o entendimento por pessoas cegas ou com baixa visibilidade. Neste aspecto, diversos trabalhos na literatura buscam estudar maneiras para facilitar a exploração e navegação em fórmulas matemáticas por meio de leitores de telas.

Ainda não há consenso na literatura sobre qual o melhor padrão que se deve seguir para a leitura efetiva de conteúdo matemático, particularmente para o idioma português do Brasil, para pessoas com deficiência visual. Segundo Lima et al. (2019), não existe um padrão definido, há variações no modo em que o mesmo conteúdo pode ser lido.

Recentemente, alguns trabalhos apresentaram avaliações relacionadas a usabilidade de leitores de telas. Neste contexto destacam-se os trabalhos de Silva et al. (2017) e de Silva et al. (2018) que avaliaram, através dos mesmos modelos descritos na seção 2.2, os leitores de telas JAWS, NVDA com o Mathplayer e uma versão do ChromeVox denominada NavMatBR.

A versão ChromeVox NavMatBR⁵ é uma implementação de novos recursos para o ChromeVox, desenvolvido como projeto de pesquisa do núcleo AL-CANCE (Núcleo de Pesquisas em Acessibilidade, Usabilidade, Linguística Computacional) do departamento de Ciência da Computação da UFLA.

Estes artigos buscaram avaliar o esforço necessário, por parte dos usuários, ao realizarem determinadas tarefas relacionadas à leitura de conteúdos matemáticos utilizando leitores de tela. Como resultado, os trabalhos apresentaram a medida de esforço em termos de tempo gasto para a realização das tarefas.

Ambos artigos utilizaram o modelo GOMS-KLM para a modelagem das tarefas e análise de tempo de execução das mesmas. A título de visualização, vê-se na Tabela 2.1 os tempos obtidos ao realizarem o estudo com os leitores de telas,

⁵ Disponível em <<https://bitbucket.org/ManolloGuedes/navmatbr/src/master/>>

cada linha representa o tempo de cada tarefa, já na última linha do quadro uma totalização dos mesmos. Os tempos estão contabilizados em segundos (SILVA et al., 2018).

Tabela 2.1 – Tempos de estimativa de esforço

<i>Tarefa</i>	<i>ChromeVox</i>	<i>ChromeVoxNavMatBR</i>	<i>NVDA</i>	<i>JAWS</i>
T_1	107.48s	76.04s	100.08s	131.64s
T_2	111,36s	83.96s	96.64s	152.76s
T_3	128s	100.6s	108.55s	129.96s
<i>Total</i>	346,84s	260,6s	305,27s	414,36s

Fonte: (SILVA et al., 2018)

Analisando a Tabela 2.1, é possível observar a diferença considerável de tempo tanto entre tarefas quanto no tempo total das análises. Este fato pode indicar o impacto que a escolha da abordagem de navegação utilizada pelo leitor de telas tem sobre o esforço para a realização de uma tarefa.

No trabalho de Silva et al. (2018), houve a análise do projeto ChromeVox-NavMatBR com implementações mais interativas que permitia aos usuários a realização de saltos entre elementos específicos como frações e somatórios. Com tais implementações disponibilizadas o ChromeVox-NavMatBR possibilitou maior dinâmica na navegação, que por sua vez possibilitou um ganho de tempo expressivo quando comparado aos demais navegadores.

Apesar de que o JAWS é bem difundido e utilizado, os resultados obtidos nas análises dos trabalhos anteriores com o mesmo não foram tão satisfatórios quando comparado aos demais leitores de telas.

Nos trabalhos de Silva et al. (2017) e Silva et al. (2018), porém, a avaliação utilizando o NVDA foi feita tendo como base sua navegação linear. Não contemplando, portanto, a utilização de *plugins* que possibilitassem uma navegação estrutural nas fórmulas matemáticas.

Como exemplo de *plugin* para navegação estrutural utilizando o NVDA temos o Access8Math. No âmbito deste trabalho, incluiremos nas análises duas

versões do plugin Access8Math, uma versão com navegação puramente em árvore e outra versão contando com a navegação contextual.

3 METODOLOGIA

Neste capítulo é apresentada a metodologia utilizada para desenvolver este trabalho de conclusão de curso. A Seção 3.1 apresenta o desenho do estudo realizado. Na Seção 3.2 é descrita uma análise do plugin. Na Seção 3.3 destaca-se como foi feita a tradução para a versão português do Brasil e, por fim, a Seção 3.4 apresenta o método para análise a análises de tarefas e estimativa de esforço.

3.1 Desenho do estudo

O objetivo deste trabalho foi de realizar uma análise de esforço na utilização do leitor de telas NVDA em conjunto com o *plugin* Access8Math. Para isso, a modelagem das tarefas e estimativa de tempo foram feitas seguindo o modelo GOMS-KLM.

Com a finalidade de estimar o tempo, o modelo GOMS-KLM foi utilizado com três operadores, sendo eles: K para o pressionamento de teclas que possui valor de 0.28 segundos, M para o processamento mental por parte do usuário que possui valor 1.2 segundos e W para aguardar resposta do sistema que possui valor de 1 segundo.

O operador W representa essencialmente a leitura de palavras por parte do leitor de telas, ao qual foi atribuído o valor médio de um segundo para cada palavra lida. O tempo total para executar uma tarefa é equivalente à soma dos três operadores descritos acima, ou seja, tempo total para executar = $0.28k + 1.2M + 1W$. Os valores dos operadores seguem o valores definidos no trabalho de Silva et al. (2018).

Foram três modelos para cada uma das 3 tarefas descritas com o modelo GOMS-KLM limitado a condições em que o usuário já saiba quais são as tarefas, o que se deve fazer e possuir conhecimento prévio em matemática. As tarefas são relacionadas a navegação em conteúdo matemático web, estruturado em MathML,

cada uma delas atrelada a uma expressão matemática diferente, como somatórios ou expressões algébricas.

O primeiro modelo das tarefas considerou a utilização do Access8Math de uma forma mais ingênua, possibilitando realizar as metas e atingir o objetivo de uma forma mais demorada, uma vez que ao encontrar a expressão desejada o *plugin* repetia a leitura do mesmo ao adentrar no modo interação.

O segundo modelo possibilitou uma abordagem em que o usuário, através das teclas de navegação, impediria a repetição do conteúdo ao entrar no modo interação, isto evitou que o mesmo escutasse novamente a pronúncia da fórmula, que por sua vez acarretou em ganho de tempo.

Para a realização do terceiro modelo, foi instalado no ambiente de testes uma nova versão do *plugin* que implementa recursos de navegação baseadas no modo contextual. O objetivo desta abordagem é possibilitar vantagens semelhantes ao do segundo modelo e proporcionar ganho de tempo quanto a pronúncia dos elementos matemáticos contidos na fórmula.

Ao final da aplicação do modelo GOMS-KLM para a realização das tarefas, obteve-se resultados do somatório dos operadores para cada tarefa e um total para cada modelo. Não foi necessário realizar a aplicação do modelo com usuários reais.

Ressalta-se que as métricas de tempo apresentadas nas próximas seções permitem comparações e análises, porém não refletem o real tempo que os usuários levariam para executar as tarefas.(SILVA et al., 2018).

3.2 Análise do plugin Access8Math

Ao avaliar a possibilidade de realizar o estudo com o Access8Math, inicialmente foi feita uma leitura da documentação¹ e do código² na linguagem Python

¹ Disponível em <https://addons.nvda-project.org/addons/access8math.pt_BR.html>

² Disponível em <<https://github.com/tsengwoody/Access8Math/tree/master/addon>>

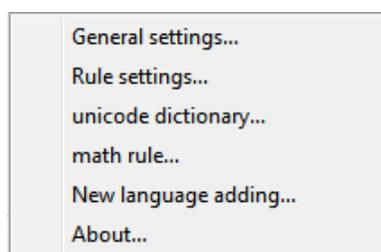
disponibilizado pelo desenvolvedor. É importante ressaltar que o *plugin* possui código aberto, o que facilita o acesso e a realização de estudos sobre ele. Neste contexto foi montado um ambiente virtual utilizando como sistema operacional o Windows 7 e navegador web Mozilla Firefox para realização de testes iniciais com a ferramenta.

Os testes iniciais objetivaram entender a navegação do plugin Access8Math. Uma página web³ contendo alguns exemplos de notações matemáticas disponíveis foi utilizada para tais testes.

O *plugin* possui algumas propriedades em sua interface que permitem alterar o idioma do sintetizador de voz. Entretanto, o *plugin* não possui muitos idiomas disponíveis, são eles: inglês, francês e chinês. Utilizou-se inicialmente a língua inglesa para os testes de leitura de conteúdo matemático.

A Figura 3.1 apresenta um captura de tela contendo as opções de configuração disponibilizadas pelo *plugin*.

Figura 3.1 – Menu de configurações do Access8Math



Fonte: do Autor 2020.

A versão disponibilizada por Guedes (2019) conta com uma nova funcionalidade de navegação baseada na estratégia de navegação contextual. Nesta versão para ativar a funcionalidade de interação utiliza-se as teclas 'CTRL' + 'ALT' + 'A', as demais funções de navegação permaneceram inalteradas.

³ Disponível em <https://github.com/ManolloGuedes/Access8Math/tree/abstract_navigation>

3.3 Tradução da leitura e interface do Access8Math

A estrutura do projeto está subdividida em três diretórios principais, sendo eles a documentação que pode abranger diversos idiomas, a pasta local que contém arquivos de mensagem que são utilizadas em algumas interações e a pasta *global plugins* diretório principal onde fica armazenado os principais arquivos do projeto, desde a arquivos para inicialização até arquivos de lógica do funcionamento do mesmo.

Na pasta principal também ficam armazenados arquivos que permitem realizar a tradução de como elementos matemáticos serão lidos, como exemplo uma fração. Também fica guardado neste diretório os arquivos sobre a pronúncia de símbolos matemáticos. Ambos arquivos podem ser disponibilizados em diversos idiomas.

Ao analisar a estruturação dos arquivos no diretório do plugin, observou-se que poderia ser feito uma tradução tanto das regras matemáticas quanto dos símbolos matemáticos, uma vez que o estudo tem por objetivo a análise baseada na língua portuguesa.

O arquivo que contém as regras matemáticas é chamado "math.rule" e o que contém os símbolos "unicode.dic". Manualmente criou-se novos arquivos já traduzidos, um para as regras matemáticas e outro para os símbolos, ambos foram importados na interface do Access8Math, o que permitiu uma leitura de conteúdo matemático pelo sintetizador de voz em língua portuguesa do Brasil.

Tais arquivos que foram alterados, estão estruturados em texto com tabulação, onde cada linha do arquivo representa uma regra matemática ou um símbolo. Para melhor esclarecer, segue um exemplo de algumas regras na figura 3.2 e de alguns símbolos na Figura 3.3.

Com o intuito de testar as traduções para verificar se as traduções estavam sendo pronunciadas pelo sintetizador de voz, importou-se os novos arquivos no menu do Access8Math, descrito na Figura 3.1.

Figura 3.2 – Arquivo math.rule

```

+ x math.rule
1 #
2 node , (.), item nó
3 mfrac fração com numerador, 0, e denominador, 1, final da fração numerador,denominador fração genérica
4 mfncced parênteses abertos, (.), parênteses fechados item parênteses genéricos
5 msqrt raiz quadrada de, (.), final da raiz item raiz quadrada

```

Fonte: do Autor 2020.

Figura 3.3 – Arquivo unicode.dic

```

+ x unicode.dic
29 α letra grega alfa none
30 β letra grega beta none
31 γ letra grega gama none
32 δ letra grega delta none

```

Fonte: do Autor 2020.

Evidentemente, a leitura de um conteúdo matemático pode ser feita de diversas maneiras. A fim de encontrar uma padronização para a leitura matemática para pessoas com deficiência visual, o estudo de Lima et al. (2019) analisou as diversas formas de leitura de conteúdo matemático e propõe que quanto mais completa a leitura melhor para o entendimento.

Por essa razão, ao traduzir os arquivos deste plugin, buscou-se manter a linguagem mais completa das expressões e símbolos matemáticos, seguindo a abordagem levantada no projeto ChromeVox-NavMatBR do núcleo ALCANCE (ABREU et al., 2019). Entretanto, quanto a entonação da voz do sintetizador nada foi alterado.

3.4 Análise de tarefas e estimativa de esforço

Considerando o vasto universo matemático, diversas fórmulas e notações podem ser criadas para expressar uma relação matemática, como somatórios, frações e raízes. Com o propósito de criar tarefas para realização do estudo, foram selecionadas três notações distintas, descritas nas equações 3.1, 3.2 e 3.3.

$$2x \times -\frac{1}{16} + -5^2 = 24\frac{15}{16} \quad (3.1)$$

$$\sum_{m=1}^3 \sum_{n=1}^m \sin(x^m + y^n) \quad (3.2)$$

$$A = \sqrt{12^2} + \sqrt[3]{125} \quad (3.3)$$

A Tabela 3.1 descreve as tarefas estabelecidas para serem realizadas sobre as fórmulas matemáticas apresentadas anteriormente. Com o intuito de possibilitar comparações de resultados, as equações, as tarefas e as métricas do KLM descritas na Tabela 3.2, foram baseadas no artigo de Card, Moran e Newell (1980).

Tabela 3.1 – Tarefas

Tarefa	Descrição
1	Verificar se a expressão possui numerador cujo valor é quinze, na fórmula 3.1
2	Verificar o limite superior no somatório, no somatório 3.2
3	Verificar o índice da segunda raiz na expressão 3.3

Fonte: (SILVA et al., 2018)

Com as expressões matemáticas e as tarefas a serem realizadas pode-se criar a modelagem de tarefas em forma de pseudo-código utilizado o GOMS, e para cada passo no modelo atribuir o valor dos operadores no KLM.

Tabela 3.2 – Variáveis KLM

Variável	Descrição
K	Representa o tempo de pressionamento de teclas
M	Representa o tempo de processamento mental do usuário
W	Representa o tempo de espera do sistema (leitura de palavras)

Fonte: (CARD; MORAN; NEWELL, 1980)

Para a realização dos testes para as análises, foi criada uma página em HTML com as notações matemáticas em MathML. A página apresenta apenas o título de cada expressão matemática seguida da fórmula a ser lida, totalizando 3 elementos de título e 3 fórmulas matemáticas. A Figura 3.4 apresenta a página utilizada na avaliação.

Figura 3.4 – Esboço da página HTML criada para os testes

Equação complexa

$$2x \times -\frac{1}{16} + -5^2 = 24\frac{15}{16}$$

Somatórios

$$\sum_{m=1}^3 \sum_{n=1}^m \sin x^m + y^n$$

Raízes

$$A = \sqrt{12^2} + \sqrt[3]{125}$$

Fonte: Página HTML criada para testes

Com as notações definidas, o objetivo de cada tarefa, as variáveis do KLM e a página HTML criada, fica possível começar a executar manualmente as tarefas de cada modelo a fim de gerar o resultados, que serão discutidos no próximo capítulo.

4 RESULTADOS E DISCUSSÃO

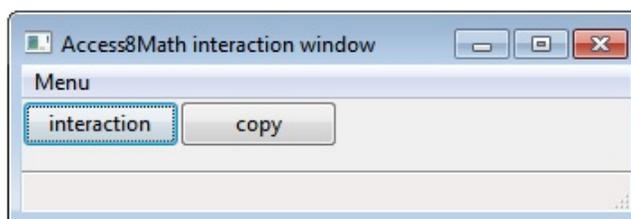
Este capítulo apresenta os principais resultados obtidos no trabalho e discussão.

4.1 Análise do Access8Math

O plugin Access8Math foi utilizado em duas versões: a versão original do mesmo disponibilizada no site do desenvolvedor¹ que utiliza navegação em árvore. A outra versão implementada no trabalho de Guedes (2019), que traz um novo modo de navegação no conteúdo matemático, este modo utiliza a navegação contextual.

Os comandos disponíveis no NVDA permitem a navegação e leitura na página web, entretanto a leitura do conteúdo matemático fica a cargo de terceiros, neste caso o *plugin* Access8Math. Na versão original, ao encontrar um conteúdo matemático web a leitura é feita de modo contínuo e linear. Para que seja possível navegar no conteúdo matemático, é necessário pressionar a tecla espaço do teclado, ativando assim o menu de interação como visto na Figura 4.1, há um aviso em voz indicando que este modo está ativado.

Figura 4.1 – Menu de interação para conteúdo matemático



Fonte: Access8Math

Por padrão o botão "*interaction*" fica selecionado, bastando pressionar a tecla espaço novamente para ativar o modo interação. Os possíveis comandos do

¹ Disponível em <https://addons.nvda-project.org/addons/access8math.pt_BR.html>.

plugin, com o menu de interação ativado para navegação, são descritos na Tabela 4.1.

Tabela 4.1 – Comandos na versão original do Access8Math

Tecla	Ação
Espaço	Ativa o modo interação
Seta para Baixo	Amplia uma parte menor do conteúdo matemático
Seta para Cima	Sai da ampliação para uma subparte maior do conteúdo matemático
Seta para a Esquerda	Passa para o elemento matemático anterior
Seta para a Direita	Passa para o próximo elemento matemático
Tecla Home	Volta para o início do conteúdo matemático
Ctrl + c	permite copiar o conteúdo matemático

Fonte: Documentação Access8Math ²

A versão original separa cada item do conteúdo matemático estruturado em MathML em elementos, por exemplo, uma fração, um sinal de operação, uma raiz ou um somatório. Ao ativar o modo interação nesta versão, a navegação pela fórmula matemática se dá item por item, lendo o conteúdo por completo de cada item.

Já na versão em que a navegação é feita pela abordagem contextual, ao ativar o modo interação o menu descrito na Figura 4.1 não é exibido na tela, um aviso por voz indica que foi ativado o modo interação, da mesma forma ocorre quando ativado o modo de leitura contextual.

Com o modo de leitura contextual ativado na nova versão disponibilizada por Guedes (2019), ao encontrar o conteúdo matemático a leitura ocorre conforme parâmetros da navegação contextual, ou seja, em um primeiro momento o leitor faz a leitura de modo linear e contínuo, porém lê-se o elemento ao invés do seu conteúdo. Dado o comando para navegar elemento por elemento da fórmula, o leitor fará a leitura do conteúdo do elemento em questão.

Os comandos disponíveis na nova versão do plugin³ estão descritos na tabela 4.2, observa-se que grande parte dos comandos mantiveram o padrão da versão original e foi acrescido um novo atalho para ativar o modo de leitura contextual.

Tabela 4.2 – Comandos na versão disponibilizada por Guedes (2019)

Tecla	Ação
Ctrl + ALT + A	Ativa o modo de leitura contextual
Espaço	Ativa o modo interação
Seta para Baixo	Amplia uma parte menor do conteúdo matemático
Seta para Cima	Sai da ampliação para uma subparte maior do conteúdo matemático
Seta para a Esquerda	Passa para o elemento matemático anterior
Seta para a Direita	Passa para o próximo elemento matemático
Tecla Home	Volta para o início do conteúdo matemático
Ctrl + c	permite copiar o conteúdo matemático

Fonte: Trabalho desenvolvido por Guedes (2019)

4.2 Expectativa de esforço para navegação no Access8Math

Para realizar os testes foram criados três modelos GOMS-KLM, para cada modelo foram criadas três tarefas, totalizando nove modelos. A primeira tarefa, como visto na Tabela 3.1, é verificar se a expressão possui numerador cujo valor é quinze na notação $2x \times -\frac{1}{16} + -5^2 = 24\frac{15}{16}$, Notação 3.1 vista anteriormente.

A segunda tarefa, descrita na Tabela 3.1, Verificar o limite superior no somatório, $\sum_{m=1}^3 \sum_{n=1}^m \sin(x^m + y^n)$, notação 3.2 vista anteriormente. Já a terceira tarefa consiste em verificar o índice da segunda raiz da expressão $A = \sqrt{12^2} + \sqrt[3]{125}$, denotada na Notação 3.3.

Cada modelo recebe um nome de acordo com seu contexto. O primeiro modelo é chamado 'leitura original'. O segundo modelo é chamado de 'evitar a

³ Disponível em <https://github.com/ManolloGuedes/Access8Math/tree/abstract_navigation>.

leitura repetida'. Já o terceiro modelo ficou nomeado como 'navegação contextual'. Ressalta-se que para a realização das tarefas no modelo leitura original e leitura repetida é utilizada a versão original do Access8Math, já o modelo navegação contextual é utilizada a versão disponibilizada através do trabalho de (GUEDES, 2019).

No modelo leitura original foram utilizados os comandos disponibilizados pela versão original do *plugin*, retratados na Tabela de comandos 4.1. Entretanto, os mesmos foram utilizados com uma abordagem ingênua caracterizada pela repetição da notação matemática ao entrar no modo de interação.

O modelo evitar a leitura repetida, também utilizados os comandos da versão original, tem como principal característica utilizar as teclas de navegação para evitar a leitura repetida da notação ao entrar no modo interação, dessa maneira a abordagem deste difere-se do modelo anterior.

As características que o terceiro modelo traz são: possibilitar uma nova maneira de interação com o conteúdo matemático através da navegação contextual e evitar a pronúncia repetida da expressão ao encontrá-la e ativar o modo interação.

Deve-se atentar ao fato de que, para acessar a terceira notação seria necessário passar pelas duas primeiras notações, ação que geraria uma adição de tempo nas tarefas 2 e 3, pois as mesmas seriam lidas novamente. Sendo assim utilizando os recursos de navegações textuais fornecidas pelo NVDA, utilizou-se a tecla H para navegar entre os títulos, evitando repetir as notações que não fariam parte da tarefa. Uma característica ofertada pelo plugin é que as teclas de comandos remetem a comandos do NVDA que são utilizados para navegação em objetos.

As Figuras 4.2, 4.3 e 4.4, são as descrições das modelagens de tarefa um em pseudo-código com o modelo GOMS-KLM para a leitura original. Já as Figuras 4.5, 4.6 e 4.7 são retratadas as tarefas para o modelo evitar a leitura repetida. As demais Figuras 4.8, 4.9, 4.10 são demonstradas as tarefas utilizando navegação contextual.

Figura 4.2 – Modelo Leitura original - GOMS para a tarefa 1 usando o Access8Math

OBJETIVO: VERIFIQUE SE A EXPRESSÃO TEM NUMERADOR DE VALOR QUINZE
 META: ENCONTRE A EXPRESSÃO DESEJADA
 SETA PARA BAIXO 1K + W (2) + 1M
 SETA PARA BAIXO 1K + W (17) + 1M
 META: ENTRAR NO MODO INTERAÇÃO
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA ESPAÇO 1K + W (17) + 1M
 VERIFICAR SE O VALOR É QUINZE
 SETA PARA BAIXO 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (6) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (4) + 1M
 SETA PARA DIREITA 1K + W (5) + 1M
 TAREFA: 14K + W(80) + 14M = 100.72

Fonte: Teste com Access8Math, do autor 2020

Figura 4.3 – Modelo Leitura original - GOMS para a tarefa 2 usando o Access8Math

OBJETIVO: VERIFICAR O LIMITE SUPERIOR DO SOMATÓRIO
 META: ENCONTRE A EXPRESSÃO DESEJADA
 TECLA H 1K + W (5) + 1M
 SETA PARA BAIXO 1K + W (42) + 1M
 META: ENTRAR NO MODO INTERAÇÃO
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA ESPAÇO 1K + W (50) + 1M
 VERIFICAR O LIMITE SUPERIOR DO SOMATÓRIO
 SETA PARA BAIXO 1K + W (16) + 1M
 SETA PARA DIREITA 1K + W (16) + 1M
 TAREFA: 6K + W(137) + 6M = 145.88

Fonte: Teste com Access8Math, do autor 2020

Figura 4.4 – Modelo Leitura original - GOMS para a tarefa 3 usando o Access8Math

OBJETIVO: VERIFICAR O ÍNDICE DA SEGUNDA RAIZ NA EXPRESSÃO
 META: ENCONTRE A EXPRESSÃO DESEJADA
 TECLA H 1K + W (5) + 1M
 SETA PARA BAIXO 1K + W (20) + 1M
 META: ENTRAR NO MODO INTERAÇÃO
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA ESPAÇO 1K + W (28) + 1M
 VERIFICAR O ÍNDICE DA RAIZ
 SETA PARA BAIXO 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (10) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (13) + 1M
 TAREFA: 9K + W(93) + 9M = 106.32

Fonte: Teste com Access8Math, do autor 2020

Figura 4.5 – Modelo Evitar leitura repetida - GOMS para a tarefa 1 usando o Access8Math

OBJETIVO: VERIFIQUE SE A EXPRESSÃO TEM NUMERADOR DE VALOR QUINZE
 META: ENCONTRE A EXPRESSÃO DESEJADA
 SETA PARA BAIXO 1K + W (2) + 1M
 SETA PARA BAIXO 1K + W (17) + 1M
 META: ENTRAR NO MODO INTERAÇÃO E POSICIONAR NO INÍCIO DA EXPRESSÃO
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA PARA CIMA 1K + W (2) + 1M
 VERIFICAR SE O VALOR É QUINZE
 SETA PARA BAIXO 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (6) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (4) + 1M
 SETA PARA DIREITA 1K + W (5) + 1M
 TAREFA: 14K + W(65) + 14M = 85.72

Fonte: Teste com Access8Math, do autor 2020

Figura 4.6 – Modelo Evitar leitura repetida - GOMS para a tarefa 2 usando o Access8Math

OBJETIVO: VERIFICAR O LIMITE SUPERIOR DO SOMATÓRIO
 META: ENCONTRE A EXPRESSÃO DESEJADA
 TECLA H 1K + W (5) + 1M
 SETA PARA BAIXO 1K + W (42) + 1M
 META: ENTRAR NO MODO INTERAÇÃO E POSICIONAR NO INICIO DA EXPRESSÃO
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA PARA CIMA 1K + W (2) + 1M
 VERIFICAR O LIMITE SUPERIOR DO SOMATÓRIO
 SETA PARA BAIXO 1K + W (16) + 1M
 SETA PARA DIREITA 1K + W (16) + 1M
 TAREFA: $6K + W(89) + 6M = 97.88$

Fonte: Teste com Access8Math, do autor 2020

Figura 4.7 – Modelo Evitar leitura repetida - GOMS para a tarefa 3 usando o Access8Math

OBJETIVO: VERIFICAR O ÍNDICE DA SEGUNDA RAIZ NA EXPRESSÃO
 META: ENCONTRE A EXPRESSÃO DESEJADA
 TECLA H 1K + W (5) + 1M
 SETA PARA BAIXO 1K + W (20) + 1M
 META: ENTRAR NO MODO INTERAÇÃO E POSICIONAR NO INICIO DA EXPRESSÃO
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA PARA CIMA 1K + W (2) + 1M
 VERIFICAR O ÍNDICE DA RAIZ
 SETA PARA BAIXO 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (10) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (13) + 1M
 TAREFA: $9K + W(67) + 9M = 80.32$

Fonte: Teste com Access8Math, do autor 2020

4.3 Comparação da expectativa de esforço do Access8Math com outros leitores de tela

Para realizar o estudo foi utilizado as mesmas técnicas de estruturação e análise de tarefas descritas no trabalho Silva et al. (2018) assim como as notações

Figura 4.8 – Modelo Navegação contextual - GOMS para a tarefa 1 usando a nova versão Access8Math

OBJETIVO: VERIFIQUE SE A EXPRESSÃO TEM NUMERADOR DE VALOR QUINZE
 META: ENCONTRE A EXPRESSÃO DESEJADA
 SETA PARA BAIXO 1K + W (2) + 1M
 SETA PARA BAIXO 1K + W (14) + 1M
 META: ENTRAR NO MODO INTERAÇÃO ABSTRATO E POSICIONAR NO INICIO DA EXPRESSÃO
 TECLA CTRL + ALT + A 3K + W (4) + 1M
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA PARA ESQUERDA 1K + W (2) + 1M
 VERIFICAR SE O VALOR É QUINZE
 SETA PARA BAIXO 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (5) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (4) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (4) + 1M
 SETA PARA DIREITA 1K + W (4) + 1M
 SETA PARA DIREITA 1K + W (5) + 1M

 TAREFA: 17K + W(56) + 17M = 78.16

Fonte: Teste com Access8Math, do autor 2020

matemáticas, com isso é possível realizar uma comparação de resultados deste trabalho com o realizado, que contemplou 4 leitores de telas diferentes.

As colunas 'AC1', 'AC2' e 'AC3' na Tabela 4.3, representam, respectivamente, os tempos de execução obtidos com os modelos de tarefas: 'Leitura original', 'Evitar leitura repetida' e 'Navegação contextual'. A Tabela 4.3 é uma adaptação da Tabela 2.1, vista anteriormente.

Observando cada linha da Tabela 4.3 e a coluna 'AC1', que demonstram os resultados obtidos com o primeiro modelo das tarefas, temos que na tarefa 1 o Access8Math obteve vantagem de aproximadamente 7 segundos ao tempo de execução da mesma no ChromeVox, aproximadamente 25 segundos mais lento do

Figura 4.9 – Modelo Navegação contextual - GOMS para a tarefa 2 usando a nova versão Access8Math

OBJETIVO: VERIFICAR O LIMITE SUPERIOR DO SOMATÓRIO
 META: ENCONTRE A EXPRESSÃO DESEJADA
 TECLA H 1K + W (5) + 1M
 SETA PARA BAIXO 1K + W (39) + 1M
 META: ENTRAR NO MODO INTERAÇÃO ABSTRATO E POSICIONAR NO INICIO DA EXPRESSÃO
 TECLA CTRL + ALT + A 3K + W (4) + 1M
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA PARA ESQUERDA 1K + W (2) + 1M
 VERIFICAR O LIMITE SUPERIOR DO SOMATÓRIO
 SETA PARA BAIXO 1K + W (14) + 1M
 SETA PARA DIREITA 1K + W (14) + 1M
 TAREFA: 9K + W(89) + 9M = 99.32

Fonte: Teste com Access8Math, do autor 2020

Figura 4.10 – Modelo Navegação contextual - GOMS para a tarefa 3 usando a nova versão Access8Math

OBJETIVO: VERIFICAR O ÍNDICE DA SEGUNDA RAIZ NA EXPRESSÃO
 META: ENCONTRE A EXPRESSÃO DESEJADA
 TECLA H 1K + W (5) + 1M
 SETA PARA BAIXO 1K + W (6) + 1M
 META: ENTRAR NO MODO INTERAÇÃO ABSTRATO E POSICIONAR NO INICIO DA EXPRESSÃO
 TECLA CTRL + ALT + A 3K + W (4) + 1M
 TECLA ESPAÇO 1K + W (8) + 1M
 TECLA PARA ESQUERDA 1K + W (2) + 1M
 VERIFICAR O ÍNDICE DA RAIZ
 SETA PARA BAIXO 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (10) + 1M
 SETA PARA DIREITA 1K + W (3) + 1M
 SETA PARA DIREITA 1K + W (13) + 1M
 TAREFA: 10K + W(57) + 10M = 71.8

Fonte: Teste com Access8Math, do autor 2020

que se fosse executada no ChromeVox NavMatBR e quando feito com NVDA o tempo foi similar. Obteve uma vantagem de 30 segundos em relação ao JAWS.

Na tarefa 2 houve aproximadamente 35 segundos de lentidão em relação ao ChromeVox, 61 em relação ao ChromeVox NavMatBR e 49 em relação ao NVDA. Em relação ao JAWS obteve vantagem de 6 segundos. Já na terceira tarefa houve vantagem de aproximadamente 22 segundos em relação ao ChromeVox e 24 em relação ao JAWS. Em relação ao ChromeVox NavMatBR foram 6 segundos de lentidão e 2 se comparado ao NVDA. Analisando os totais, há desvantagem de 6, 92 e 50 segundos para o ChromeVox, ChromeVox NavMatBR e NVDA respectivamente. Houve vantagem de 61 segundos em relação ao JAWS.

Para o modelo leitura original, vê-se que em cada vez que as linhas das tarefas que remetem à abertura do menu de interação são realizadas há em seguida um acréscimo de tempo pois tal menu, além de ter o aviso em áudio de uma frase indicando que o mesmo está ativado, faz também a releitura da notação matemática em questão, o que por sua vez adicionou 25 segundos na primeira tarefa, 58 segundos na segunda e 36 segundos na terceira.

Ao observar a coluna 'AC2' que compete ao modelo 'Evitar leitura repetida' para as tarefas, nota-se que a execução da tarefa 1 obteve vantagem de aproximadamente 22, 15 e 46 segundos quando comparados com ChromeVox, NVDA e JAWS respectivamente. Em relação ao ChromeVox NavMatBR houve desvantagem de 9 segundos.

Já na tarefa 2 no modelo 'Evitar leitura repetida', observado na tabela 4.3, obteve-se os seguintes resultados: 14 e 54 segundos de vantagem quando comparado com ChromeVox e ao JAWS. Houve desvantagem de 1 segundo em relação ao NVDA e 14 ao ChromeVox NavMatBR. Para a tarefa 3 houve vantagem em relação a todos os outros leitores de telas, sendo aproximadamente 48, 20, 28 e 49 segundos para ChromeVox, ChromeVox NavMatBR, NVDA e JAWS respectivamente.

Em relação aos totais das tarefas executadas no modelo 'Evitar leitura repetida', atingiu-se a vantagem de aproximadamente 83, 50 e 150 segundos em

relação ao ChromeVox, NVDA e JAWS respectivamente. A desvantagem em relação ao ChromeVox NavMatBR foi de apenas 3 segundos.

Temos que, ao considerar o pressionamento da tecla seta para cima ao adentrar no modo de interação no segundo modelo de tarefas, obteve-se um ganho expressivo de tempo em relação aos resultados obtidos por outros leitores de tela e ao primeiro modelo leitura original para as tarefas.

A coluna 'AC3' da Tabela 4.3 evidencia os tempos de estimativa de esforço ao ativar o modo de leitura contextual da nova versão disponibilizada. O modelo 'Navegação contextual', que utiliza a estratégia de leitura contextual do conteúdo matemático, obteve resultado de grande vantagem em relação ao modelo 'leitura original', um total de aproximadamente 104 segundos. Ao observar a coluna 'AC2', que diz respeito ao modelo 'evitar repetição', há vantagem total de 14,64 segundos, teve desvantagem de 3 segundos apenas na execução da segunda tarefa.

O ganho de tempo do modelo 'navegação contextual' em relação ao modelo 'evitar leitura repetida' se deve ao fato de que ao ativar este modo de leitura o *plugin* não faz a leitura na íntegra da fórmula em si, a leitura é feita por elementos, ou seja, caso tenha alguma fração, por exemplo, ao invés de ler o numerador e o denominador é dito que o elemento é uma fração em um primeiro momento, só após pressionar espaço é que o usuário pode navegar no conteúdo na íntegra.

O tempo de execução das tarefas do modelo 'navegação contextual', comparado aos resultados dos testes realizados anteriormente nos demais leitores de telas e aos modelos de tarefas 'leitura original' e 'evitar leitura repetida' realizados neste trabalho, demonstrou o menor tempo para realização das tarefas.

Outro ponto a ser considerado é que mesmo possibilitando ações de navegação no conteúdo matemático, a mesma se dá elemento por elemento. Tomemos por exemplo a tarefa 1 que por sua vez necessitava analisar o último elemento da

notação. Neste cenário houve a necessidade de navegar por todo o conteúdo item a item, o que representou um tempo considerável de navegação.

Tabela 4.3 – Tempos de estimativa de esforço de leitores de telas anteriores e os 3 modelos utilizando o Access8Math

Tarefa	Chrome Vox	Chrome Vox NavMat BR	NVDA	JAWS	AC1	AC2	AC3
<i>T1</i>	107.48s	76.04s	100.08s	131.64s	100.72s	85.72s	78.16s
<i>T2</i>	111,36s	83.96s	96.64s	152.76s	145.88s	97.88s	99.32s
<i>T3</i>	128s	100.6s	108.55s	129.96s	106.32s	80.32s	71.80s
Total	346.84s	260.6s	305.27s	414.36s	352,92s	263.92s	249.28s

Fonte: do Autor 2020, adaptado de (SILVA et al., 2018)

Ressalta-se que a versão disponibilizada do plugin que permite a navegação contextual não foram implementadas teclas de atalhos para navegação entre os elementos, diferentemente da implementação em árvore realizada no projeto ChromeVox NavMatBR, onde era possível navegar entre elementos através de novos atalhos disponibilizados. (SILVA et al., 2018).

Tais teclas de atalho possibilitariam acessar elementos diretamente, como por exemplo frações ou somatórios. Ou seja, caso o objetivo da tarefa fosse analisar um elemento matemático que estivesse ao final da fórmula, seria pressionado determinados comandos de atalho, por exemplo, para evitar a leitura de todos os elementos anteriores.

Entretanto, a disponibilização de novos atalhos poderia causar sobrecarga mental aos usuários, uma vez que a versão do Access8Math que disponibiliza o recurso de navegação contextual traz apenas um novo atalho de ativação e desligamento do modo de 'leitura contextual', mantendo os demais atalhos de navegação já familiarizados na versão original do *plugin*.

Apesar de não implementados novos atalhos para navegação entre elementos, os tempos demonstrados na Tabela 4.3 evidenciam que o modelo 'navegação contextual' obteve melhor resultado em seu tempo total de execução.

Como os testes não foram realizados com usuários reais e sim pelo autor do projeto, compreende-se que o tempo de execução de cada tarefa pode ser ainda maior quando levado em consideração o processamento mental por parte do usuário. Nas tarefas aqui realizadas foi considerado que o mesmo já tinha certa experiência com a matemática e utilização das duas versões disponibilizadas do *plugin*.

5 CONCLUSÃO

O objetivo deste trabalho foi estimar a expectativa de esforço do usuário ao utilizar o leitor de telas NVDA acrescido do *plugin* para leitura de conteúdo matemático denominado Access8Math. O método utilizado foi o modelo GOMS-KLM que possibilitou que as tarefas necessárias para atingir um determinado objetivo fossem realizadas e ao decorrer das mesmas aplicar o KLM como ferramenta para possibilitar as métricas obtidas. Esta abordagem favoreceu a elaboração do projeto sem que fossem realizados testes com usuários reais.

Ao aplicar o método utilizou-se de três modelos distintos entre si com a finalidade de criar as análises. Em um primeiro momento o uso da ferramenta de uma forma mais ingênua descrito no modelo leitura original das atividades, ao passo que no segundo evitar leitura repetida das tarefas utilizou-se do pressionamento de uma tecla que eliminava a repetição da notação que estava sendo lida pelo sintetizador de voz, o que ocasionou um ganho significativo de tempo nas atividades em relação ao primeiro modelo. Já no modelo navegação contextual foi utilizado uma nova versão do *plugin* que foi disponibilizada durante o processo de escrita do trabalho.

Com as métricas estabelecidas, observa-se que a implementação do modo contextual de navegação através do Access8Math possibilitou um ganho de 14,64 segundos em relação ao modelo evitar leitura de repetição, evidenciando que a abordagem surtiu efeitos positivos nas análises.

Fica evidenciado então que o *plugin* Access8Math em sua versão que possibilita a utilização do modo de leitura contextual pode obter vantagem em relação aos demais recursos de navegação disponíveis e analisados em trabalhos anteriores, uma vez que o tempo de estimativa de esforço foi o menor em relação aos demais.

Apesar das limitações apresentadas no modelo GOMS-KLM, onde as tarefas não são realizadas com usuários reais, o estudo permite auxiliar no aponta-

mento de sugestões para melhoria a serem acrescentadas no *plugin* com a finalidade de possibilitar que o ensino e aprendizagem da matemática seja mais eficiente aos usuários com deficiência visual, reduzindo a carga mental e tornando o processo de aprendizado cada vez mais eficiente.

Evidentemente, testes com usuários reais utilizando outros métodos de análise proporcionariam uma visão mais incisiva em relação às possíveis melhorias que poderiam ser implementadas tanto no Access8Math como em outros navegadores de tela que possuem suporte para leitura de conteúdo matemático ou na WEB ou em editores de texto.

Possibilitar novos atalhos ou a combinação dos mesmos para a navegação pode incidir em um aumento de carga mental, em contra partida pode aumentar as maneiras de interação de usuários com o conteúdo matemático, permitindo acessar elementos de formas diferentes.

É importante acrescentar que este trabalho de conclusão de curso fez parte de um trabalho maior envolvendo a análise e desenvolvimento de uma nova metodologia de navegação, bem como aplicação da nova metodologia a um software de leitura de telas. Como resultado deste trabalho macro de pesquisa já obtivemos a nova versão do *plugin* Access8Math, que contou com a leitura utilizando navegação contextual e abstração de conteúdo complexo, proposto inicialmente por (STEVENS; EDWARDS; HARLING, 1997).

Como trabalhos futuros, serão realizadas melhorias na navegação da nova versão do *plugin*, tendo como base as análises levantadas por este trabalho de conclusão de curso. Além disso, serão feitos testes com usuários com deficiência visual para a validação da nova proposta de navegação, bem como identificação de pontos de impacto e de melhoria. Posteriormente, a versão final do *plugin* será disponibilizada gratuitamente para uso.

REFERÊNCIAS

ABREU, S. et al. Usability evaluation of a resource to read mathematical formulae in a screen reader for people with visual disabilities. In: **Proceedings of the 18th Brazilian Symposium on Human Factors in Computing Systems**. Vitória, ES: ACM, 2019. p. 1–11.

CARD, S.; MORAN, T.; NEWELL, A. The keystroke-level model for user performance time with interactive systems. **Commun. ACM**, v. 23, p. 396–410, 07 1980.

CARD, S. K.; NEWELL, A.; MORAN, T. P. **The Psychology of Human-Computer Interaction**. USA: L. Erlbaum Associates Inc., 1983. ISBN 0898592437.

COOK, A. M.; POLGAR, J. M. **Assistive Technologies: Principles and Practice**. 4. ed. [S.l.]: Elsevier, 2014.

EVERIS BRASIL. **Pesquisa brasileira do uso de leitores de tela**. 2019. Disponível em: <https://mwpt.com.br/wp-content/uploads/2019/07/Pesquisa-LDT_Relatorio.pdf>. Acesso em: 29 de abril de 2020.

FAULKNER, S. et al. **HTML 5.2**. [S.l.]: <https://www.w3.org/TR/2017/REC-html52-20171214/>, 2017.

GAURA, P. Remathex — reader and editor of the mathematical expressions for blind students. In: **Proceedings of the Internacional Conference Computers Helping People with Special Needs (ICCHP)**. Linz, Austria: Springer, 2002. v. 2398, p. 486–493.

GUEDES, H. M. C. **Estratégias de navegação em fórmulas matemáticas na WEB para pessoas com deficiência visual**. Lavras: Universidade Federal de Lavras, 2019. Monografia de qualificação de mestrado. Programa de Pós-Graduação em Ciência da Computação. Universidade Federal de Lavras.

LIMA, M. A. de et al. Análise de verbalizações de fórmulas matemáticas por professores com experiência no ensino de pessoas com deficiência visual/analysis of mathematical formulas verbalizations by teachers with experience in teaching visually impaired people. **Revista de Estudos da Linguagem**, v. 27, n. 3, p. 1371–1397, 2019.

MINER, R. R.; ION, P. D. F.; CARLISLE, D. **Mathematical Markup Language (MathML) Version 3.0 2nd Edition**. [S.l.], 2014. <http://www.w3.org/TR/2014/REC-MathML3-20140410/>.

RADABAUCH, M. Technology for access and function research section two: Niddr research agenda chapter 5. **Technology for access and Function**, 1993.

RAMAN, T. **Audio System For technical Readings**. 1994. Tese (Doutorado), Cornell University, Dept. of Computer Science., Ithaca, NY, USA.

SILVA, L. F. da P. et al. Content-based navigation within mathematical formulae on the web for blind users and its impact on expected user effort. **Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion**, I, n. 1, p. 23–32, jun. 2018. Disponível em: <<https://dl.acm.org/citation.cfm?id=3218596>>.

SILVA, L. F. da P. et al. How much effort is necessary for blind users to read web-based mathematical formulae? a comparison using task models with different screen readers. **Brazilian Symposium on Human Factors in Computing Systems**, I, n. 1, p. 23–32, out. 2017. Disponível em: <<https://dl.acm.org/citation.cfm?id=3218596>>.

SORGE, V. et al. Towards making mathematics a first class citizen in general screen readers. In: **Proceedings of the 11th Web for All Conference**. New York, NY, USA: ACM, 2014. (W4A '14), p. 40:1–40:10. ISBN 978-1-4503-2651-3. Disponível em: <<http://doi.acm.org/10.1145/2596695.2596700>>.

STEVENS, R. D.; EDWARDS, A. D. N.; HARLING, P. A. Access to mathematics for visually disabled students through multi-modal interaction. **Journal Human-Computer Interaction**, v. 12, n. 1, p. 47–92, fev. 1997. Disponível em: <<https://dl.acm.org/citation.cfm?id=1462972>>.

W3C Web Accessibility Initiative. **Web Content Accessibility Guidelines (WCAG) 2.0**. 2008. Disponível em: <<https://www.w3.org/TR/WCAG20/>>.

WONGKIA, W.; NARUEDOMKUL, K.; CERCONE, N. Better access to math for visually impaired. In: **Proceedings of the IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH)**. Toronto, CA: IEEE, 2009. p. 43–48.