



Notação matemática para ambientes de aprendizado eletrônico online utilizando ferramentas de código aberto

Orlando de Andrade Figueiredo, IGCE-UNESP, orlando@rc.unesp.br
Luis Henrique de Rossi, IGCE-UNESP, luishderossi@hotmail.com
Marcus Vinícius Maltempi, IGCE-UNESP, maltempi@rc.unesp.br
Marcelo de Carvalho Borba, IGCE-UNESP, mborba@rc.unesp.br

Resumo. *Este artigo apresenta questões técnicas associadas ao uso de notação matemática na web e revisa as principais tecnologias para uso de notação matemática disponíveis atualmente. É apresentada uma proposta baseada em tecnologias abertas e gratuitas para viabilizar o uso da notação matemática em um ambiente de aprendizado eletrônico (learning management systems) baseado na web de forma a contornar os principais obstáculos do atual momento da web.*

Palavras-chave: *Internet, Notação Matemática, MathML, Learning Management System*

Mathematical notation for electronic learning environments, using open source online tools

Abstract. *This paper describes technical issues related to mathematical notation on the Web and reviews the main technologies currently in use for mathematical notation. A proposal for implementing mathematical notation in a learning management system on the Web, based on open and free technologies, is presented. The purpose is to overcome the main obstacles at the present moment of the Web.*

Keywords: *Internet, Mathematical Notation, MathML, Learning Management System*

1. Introdução

Discussões envolvendo ensino-aprendizagem e a utilização da internet vêm se intensificando em diversas áreas acadêmicas, em particular, em Educação Matemática. Com base em dados produzidos em cursos a distância destinados a professores de matemática, estudiosos têm apontado aspectos diferenciados do ponto de vista comunicacional que emergem ao se discutir matemática em um chat, por exemplo. Os participantes nestes ambientes expressam suas opiniões de modo aparentemente "desordenado", o que exige um modo diferenciado de pensamento. De modo mais

recente, pesquisas enfatizam alguns aspectos fundamentais sobre o processo de pensamento matemático oriundos do fato de que a grande maioria dos ambientes online não disponibiliza símbolos matemáticos para se discutir Cálculo ou ferramentas de desenho para se discutir geometria, por exemplo. Os tipos de representações disponíveis (símbolos, aspectos visuais, etc.) são elementos que condicionam os processos de produção de significados e conhecimentos (matemáticos).

Mediante esta inquietação em relação à utilização de notação matemática em ambientes virtuais, iniciamos um trabalho de caráter pedagógico e computacional que visa desenvolver uma ferramenta que disponibilize notações matemáticas em contextos online. Neste texto, caracterizaremos alguns pontos importantes que vêm envolvendo o processo de desenvolvimento desta ferramenta.

O objetivo deste trabalho é apresentar uma proposta para adoção de notação matemática em um *Learning Management System (LMS)* baseado na web a partir de tecnologias abertas. Grandes são os desafios para uso da notação matemática na web, em geral: não adoção de padrões por parte de todos os fabricantes de navegadores, falta de interfaces universalmente aceitas etc. Muitas soluções improvisadas têm sido criadas para contornar esses problemas de forma imediata. Em muitos casos, os usuários são submetidos a situações em que eles precisam configurar seus sistemas manualmente, ou instalar componentes adicionais que nem sempre possuem licença gratuita ou código aberto. Pretendemos que nossa proposta dispense o emprego de soluções proprietárias e minimize o trabalho de configuração por parte do usuário de um ambiente de aprendizagem (*LMS*). Partimos da realidade momentânea e atual das tecnologias efetivamente empregadas na web visando a uma solução para uso imediato por parte dos desenvolvedores de *LMS* baseados na web.

Na Seção 2, tratamos das questões técnicas associadas à notação matemática na web, em três frentes: apresentação dos símbolos matemáticos, formatos de representação e interface de entrada/edição. Na Seção 3, passamos em revista pelas principais tecnologias disponíveis no cenário atual relacionadas com a notação matemática na web e em geral. Na Seção 4 é descrita a proposta de integração de tecnologias para atender a necessidade do uso de notação matemática nos ambientes de aprendizado eletrônico para web. Por fim, na Seção 5, apresentamos a situação atual do andamento dos trabalhos e perspectivas.

2. Questões técnicas associadas à notação matemática

A web foi originalmente concebida como uma plataforma de divulgação e distribuição de documentos de hipertexto com um formato muito específico, o *HyperText Markup Language (HTML)*. A notação matemática, com seus símbolos, regras de apresentação gráfica e outras peculiaridades, não foram contemplados neste formato. Desde então, os interessados em utilizar a notação matemática na web têm procurado, com dificuldades, encontrar caminhos para torná-la um elemento naturalmente disponível nesse meio. Segundo Miner (2005), se comparado ao texto simples, a escrita de notação matemática é mais difícil de representar, editar e processar nos computadores.

Um dos caminhos que se abriram foi a adoção de tecnologias de programação para a web. A web é um meio muito atraente para a disponibilização de serviços e aplicações devido ao grande público que atinge. Concomitantemente, empresas e demais provedores de serviços começaram a demandar mecanismos que permitissem embutir suas aplicações em páginas web. Nesse processo surgiram tecnologias como *Cascade Style Sheets (CSS)*, *JavaScript*, mini-aplicativos e *Flash*.

O componente principal, em torno do qual estas questões emergem, é o navegador web (browser). Navegadores web bastante conhecidos são o *Internet Explorer* da *Microsoft*, o *Firefox* da *Mozilla Foundation* e o *Safari* da *Apple*. É o navegador que obtém os documentos *HTML* e os interpreta para realizar a apresentação final para o usuário. É também no navegador que essas novas tecnologias web operam. Portanto, para viabilizar a notação matemática na web, os navegadores precisam ser cuidadosamente considerados. No entanto, o primeiro desafio que se encontra é que os fabricantes de navegadores ou versões distintas dos mesmos navegadores podem apresentar significativas diferenças na forma de interpretar os documentos *HTML* (Figura 1).

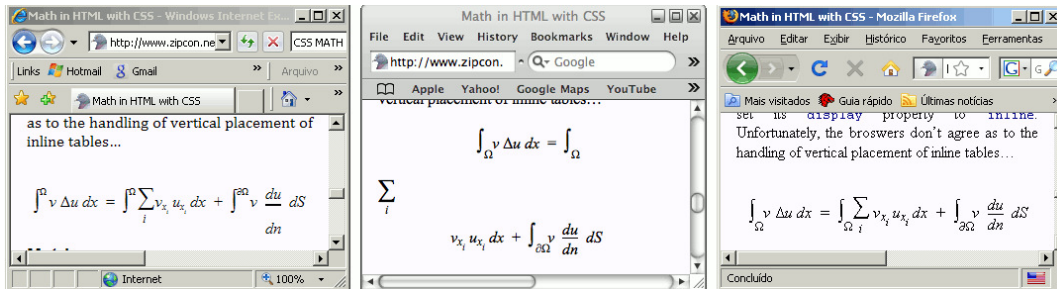


Figura 1: Conteúdo matemático exibido nos navegadores *IE*, *Safari* e *Firefox*

Nesta seção, apresentamos as principais questões técnicas que desafiam os desenvolvedores interessados em implementar notação matemática, especialmente questões relacionadas à web e aos navegadores.

2.1 Apresentações de símbolos matemáticos e sentenças

As letras, números, sinais de pontuação e outros símbolos que compõem um texto são chamados, no jargão da informática, de caracteres. Um caractere tem dois aspectos: o desenho e a codificação. O desenho é o traçado gráfico que deve ser apresentado ao usuário. A codificação é um número usado internamente para representar e armazenar o caractere. Em seu perfeito funcionamento, um programa é capaz de recuperar um texto codificado e, mediante uma tabela de codificação, encontrar os devidos desenhos que formarão a apresentação final do texto.

Existem algumas tabelas de codificação padronizadas. Os símbolos matemáticos presentes nessas tabelas restringem-se apenas aos mais corriqueiros, como ‘+’ (símbolo de adição) ou ‘=’ (símbolo de igualdade). O conjunto de todos os possíveis símbolos matemáticos é muito rico (Figura 2). Portanto, a primeira necessidade para se aplicar notação matemática em qualquer sistema informatizado é a existência de tabelas de

símbolos que contemplem uma gama mais ampla de símbolos matemáticos. Existem conjuntos específicos de símbolos para cada ramo da Matemática.

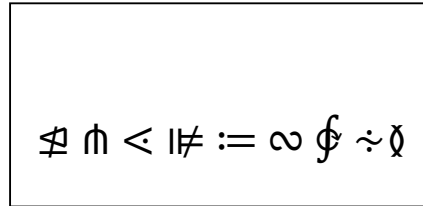


Figura 2: Exemplos de caracteres matemáticos

No caso da web, é preciso que o navegador conheça e utilize essas tabelas com símbolos matemáticos. O problema é que nem todos os navegadores trazem essas tabelas configuradas por padrão. Disso decorre que o usuário que não está familiarizado com essas questões provavelmente tenha dificuldades para trabalhar com Matemática.

Além do problema da apresentação dos símbolos, a notação matemática traz um desafio a mais em relação ao texto simples. Quando escrevemos palavras, a posição entre os caracteres é a mais direta possível: um caractere segue o outro na mesma linha. Já a notação matemática é bem mais complexa nesse aspecto. Os símbolos podem estar não só ao lado, mas em cima, em torno, inseridos uns nos outros, com diferentes tamanhos, etc. Além disso, surgem outros elementos gráficos, como barras verticais e horizontais (Figura 3). Ajustar a disposição gráfica dos elementos conforme esperado na notação matemática não é algo que os navegadores foram originalmente projetados para lidar.

$$\sqrt[3]{2 + \frac{3}{5^x}} = \det \begin{vmatrix} 5 & 0 \\ \frac{1}{2} & 3 \end{vmatrix}$$

Figura 3: Exemplo de equação usando traços para desenhar fração, raiz e matriz

2.2 Formatos para representação de sentenças matemáticas

Sentenças matemáticas têm uma estrutura complexa. Falar em estrutura é como falar na divisão de um livro. Existem regras que estabelecem como as partes internas se relacionam. No caso do livro, ele pode ser dividido em capítulos, além de índice, prefácio, notas, etc. Cada capítulo por sua vez, contém seções, itens, figuras. O texto em si encontra-se inserido nas seções. No caso da notação matemática, também existem regras que definem uma estrutura. Por exemplo, uma raiz quadrada, contém dentro de si outra sentença. Uma fração precisa de duas sub-sentenças, uma para o numerador e outra para o denominador.

Para armazenamento, processamento e transmissão de sentenças matemáticas, é imprescindível que suas estruturas sejam representadas através de formatos especiais. Da mesma forma como o navegador precisa conhecer as tabelas de caracteres com símbolos matemáticos, ele também precisa conhecer formatos de representação de

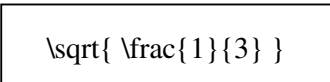
sentenças matemáticas para trabalhar com elas. Como antes, não há consenso entre os fabricantes de navegadores quanto aos formatos a serem adotados.

2.3 Interface de entrada e edição de notação matemática

A vida de quem trabalha com matemática em computadores seria muito mais fácil se os teclados possuíssem teclas para todos os símbolos matemáticos necessários. Isso talvez nunca venha a se realizar um dia, pois seriam teclados enormes, caros (por serem especiais e produzidos em pequena escala) e pouco práticos. Portanto, o desafio para o usuário de informática interessado em digitar sentenças matemáticas já se apresenta no próprio equipamento. A solução óbvia é contornar o problema por software. Duas estratégias principais dominam o cenário dos programas que lidam com notação matemática: teclados virtuais e linguagens especiais.

Teclados virtuais são conjuntos extensos de botões com tantos caracteres quantos forem necessários. A matemática é tão vasta e com tantas subdivisões, cada qual com seu jargão próprio, que essas teclas são arranjadas por área em questão (geometria, análise, álgebra, topologia, etc.), de forma que só é preciso usar um subconjunto do teclado ao mesmo tempo. Teclados virtuais são uma forma popular de entrada de sentenças matemáticas. São fáceis de aprender e usar. No entanto, quem os utiliza com grande frequência precisa recorrer ao mouse muitas vezes, o que nem sempre é produtivo e confortável.

As linguagens especiais para notação matemática são conjuntos de comandos, como numa linguagem de programação, que são entendidos como um equivalente matemático pelo sistema. Por exemplo, no popular sistema de tipografia digital *LaTeX* (ver Seção 0), o comando "\int" representa uma integral. Na Figura 4, os comandos apresentados significam a raiz quadrada de um terço. Uma linguagem especial tem muitas vantagens, como a possibilidade de uso em interfaces limitadas (computador sem modo gráfico) e por ser, ao mesmo tempo, um mecanismo de representação das sentenças matemáticas. No entanto, exige dos usuários certo treinamento e memorização de comandos.



```
\sqrt{\frac{1}{3}}
```

Figura 4: Exemplo de fórmula usando comandos *LaTeX*

A web, em sua origem, era voltada para a distribuição e leitura de documentos. Embora os documentos *HTML* já pudessem conter campos de entrada de dados simples, eles compunham formulários eletrônicos usados para obtenção de informação em aplicações. Com a evolução, a edição de textos embutidos nas páginas da web, como em aplicações de webmail, blogs, fóruns, etc., se difundiu e recursos mais sofisticados para entrada de dados surgiram. No entanto, nem os recursos de entrada de dados embutidos em páginas web, nem aqueles disponíveis nos navegadores, contemplam sentenças matemáticas sem uma preparação adicional.

3. Tecnologias relacionadas com notação matemática

Para lidar com as questões tratadas na seção anterior, podemos tomar como ponto de partida trabalhos já existentes, desenvolvidos por organizações e indivíduos interessados em matemática. Apresentamos as principais tecnologias disponíveis para a computação em geral, bem como aquelas específicas para *web*.

3.1 *LaTeX*

TeX (Knuth, 1984) é um sistema de tipografia digital completo, concebido especialmente para tratar texto com conteúdo matemático. *LaTeX* (Lamport, 1994) é uma aplicação criada usando o *TeX*. O resultado final do documento gerado é esteticamente superior em comparação com os programas mais conhecidos.

Existem diversas ferramentas que permitem utilizar *LaTeX* para viabilizar a utilização de notação matemática na *web*. A forma mais simples é aquela em que o autor da página *web* gera figuras contendo as sentenças matemáticas utilizando o *LaTeX* e as insere na página. A conversão pode ser feita por uma sistema *LaTeX* convencional ou por aplicações *web* para o mesmo fim, como é o caso do *LaTeXRender* (Mayer, 2008). Há também o caso em que a conversão é feita no navegador, utilizando um script, a exemplo do *JSMath* (Cervone, 2006) e *TrH* (Hutchinson, 2007).

Há muito material produzido em *LaTeX* e isso não pode ser ignorado no momento de se optar por um sistema de notação matemática. A linguagem de marcação do *LaTeX*, no que tange à matemática, não é semanticamente explícita e, portanto, pode ser ambígua (Fateman & Caspi, 2004)

3.2 *Unicode*

O papel das tabelas de codificação padronizadas foi discutido na Seção 0. *Unicode* (The *Unicode Consortium*, 2006) é um padrão de codificação de caracteres, relativamente recente, e que procura ambiciosamente abranger todas as possibilidades de idiomas e símbolos que são universalmente utilizados. Atualmente, o padrão *Unicode* possui uma quantidade de caracteres da ordem de 100.000. No caso particular dos símbolos matemáticos, o *Unicode* prevê uma quantidade de cerca de 3000 símbolos. Os símbolos matemáticos são agrupados em blocos de acordo com suas funções tais como setas, parênteses, operadores etc. Outros blocos de caracteres *Unicode* não especialmente voltados para matemática podem ser usados, como letras gregas e latinas.

A tendência é que o *Unicode* ganhe aceitação cada vez maior e passe a ser adotado preponderantemente pelos sistemas em geral. Os sistemas operacionais modernos e as novas versões de programas aderem ao padrão *Unicode*, incluindo os navegadores. Como já foi dito na Seção 0, atualmente o usuário não encontra esse ferramental configurado e pronto para usar. Por exemplo, embora o sistema operacional possua tabelas de codificação *Unicode*, é preciso também que alguma fonte instalada no computador possua o desenho do caractere matemático. Ou seja, é necessário conhecer os conceitos subjacentes ao sistema operacional ou contar com ajuda para poder usá-lo apropriadamente.

3.3 MathML e OpenMath

Os padrões *Mathematical Markup Language (MathML)* e *OpenMath* (Caprotti, 2000) são formatos de representação de sentenças matemáticas. O primeiro é implementado com base na linguagem *eXtensible Markup Language (XML)* e o segundo, também, mas não obrigatoriamente.

Segundo Sandhu (2002), usando o *MathML* é possível exibir notações matemáticas complexas em páginas web com alto grau de fidelidade e clareza. Ao mesmo tempo, o significado da notação pode ser preservado para, por exemplo, copiar uma equação da web e colar em um programa de álgebra para validação. O *MathML* é um padrão para representação matemática amplamente aceito que provê base para muitas aplicações úteis e interessantes. Por exemplo, é possível desenvolver páginas web dinâmicas com ferramentas para criação de equações interativas, criar uma base de dados com documentos técnicos que podem ser indexados, arquivados e buscados.

Por ser escrito no formato texto, *MathML* é portátil e independente de plataforma. Isso o torna um meio conveniente para troca de conteúdo matemático entre diversas aplicações como navegadores, editores de texto, editores de equação, bem como sistemas algébricos e outros softwares científicos. *MathML* é portanto bem adaptado para se tornar um formato padrão para armazenamento e comunicação de informação matemática, no entanto, esse processo de adoção ainda está em andamento e prossegue em passos lentos. Por exemplo, nos casos dos navegadores, o *Mozilla* trabalha com o *MathML* nativamente e o *Internet Explorer* necessita de um plugin para exibir o conteúdo matemático nas páginas.

3.4 Notação matemática em hipertexto

Como os navegadores não adotam padrões para a exibição de notação de matemática, muitos desenvolvedores passaram a improvisar soluções alternativas com base nas tecnologias web existentes e efetivamente disponíveis, tais como o formato de documento hipertexto básico (*HTML*), imagens inseridas nos documentos de hipertexto, formatação de texto com base em *CSS* etc. Todas essas estratégias visam principalmente contornar um único problema: a exibição das sentenças matemáticas. E, embora consigam satisfazer esse requisito, eles apresentam desvantagens como: entrada de dados trabalhosa e sentença representada apenas visualmente, estando indisponível para outros fins, como interoperabilidade com outros programas. Detalhes técnicos podem ser encontrados em White (2008) e Korpela (2008).

3.4.1 Imagens

Apresentar sentenças matemáticas em forma de imagem, tipicamente no formato GIF ou JPEG, é a forma mais simples e a melhor para quem quer apenas demonstrar uma fórmula pronta. Não existe incompatibilidade entre navegadores para exibição de imagens, todos conseguem exibir a imagem de forma similar, sem distorções, sem quebras e sem omitir caracteres. Por outro lado, não permite que a fórmula seja editada.



Não é possível transformar uma imagem pronta em uma nova fórmula diferente sem recorrer a programas de edição de imagens.

As imagens são geradas por programas que dão suporte à notação matemática, como *Maple*, *LaTeX*, *Mathematica*, *MS Word* (com *MS Equation*), etc. A melhor maneira de editar uma fórmula transformada em imagem é usar o software de origem para gerar uma nova imagem.

3.4.2 Sentenças montadas com caracteres não-matemáticos

Essa solução remonta ao período em que as interfaces dos computadores eram unicamente baseadas em texto. Para obter um efeito “gráfico”, como um contorno de janela ou de tabelas, montava-se um esquema aproveitando os caracteres usuais de texto, especialmente os sinais de pontuação, tais como parênteses, traços horizontais (hífen e sublinhado), traços verticais, barras, ponto etc. A Figura 5 ilustra a construção de uma sentença utilizando essa solução. O efeito visual de radical é obtido através do arranjo de diversos caracteres não matemáticos (‘\’, ‘/’, ‘-’ e ‘’). A barra de fração é um hífen. Embora o resultado seja compreensível, ele é esteticamente inferior.

$$\begin{array}{c} \text{---} \\ / 1 \text{ `} \\ / - \\ \vee 3 \end{array}$$

Figura 5: Sentença desenhada com caracteres não-matemáticos

3.4.3 Folha de Estilos

Um conjunto de recursos para formatação de texto é denominado “estilo de formatação”. Esses estilos incluem tipo de fonte, tamanho do texto, negrito, espaçamento, entre outros. Pode-se encontrar esse recurso disponível em editores de texto como o MS-Word. Na web a definição de estilos segue um padrão chamado *Cascading Style Sheets (CSS)*.

Para exibição de sentenças matemáticas, o *CSS* contribui principalmente permitindo o correto posicionamento dos elementos na fórmula e para desenho de traços usados em fração, raiz e matrizes, além de outros efeitos de formatação.

3.4.4 Mini-aplicativos

Mini-aplicativos (applets) são programas que podem ser embutidos em páginas web. Esses programas são escritos em linguagens diferentes de *HTML* e interpretados por ambientes de execução especiais (em alguns casos, chamados de máquinas-virtuais) que são instalados como complementos do navegador. Os exemplos mais conhecidos de ambientes de execução são as máquinas-virtuais Java e os plugins *Flash*.

As linguagens de programação usadas na construção dos mini-aplicativos foram projetadas para fins mais gerais do que a simples exibição de documentos *HTML*. Por isso, os mini-aplicativos proporcionam muito mais possibilidades de programação, como interatividade, capacidade de exibição de objetos gráficos complexos, variedade de mídias etc. Os ambientes de execução do mini-aplicativos não estão restritos àquilo que se passa nos navegadores, isto é, eles constituem um mundo novo e independente com que o programador pode contar para atingir seus objetivos. Assim, no caso da notação matemática, todas as etapas (formato de representação, apresentação, entrada de dados) podem ser completamente montadas desde a base em um mini-aplicativo. É claro que não faz sentido substituir completamente o navegador, a intenção é complementá-lo.

Existem editores com suporte à notação matemática totalmente implementados como mini-aplicativos. Um exemplo é o *DragMath* (Billingsley, 2008), um applet desenvolvido em *JAVA*, que funciona com o conceito de arrastar e soltar.

4. Proposta para implementação de notação matemática em um *LMS* web

Nosso problema é apontar caminhos para viabilizar o uso de notação matemática em ambientes virtuais de aprendizagem baseados na web. Apresentamos nas seções anteriores as tecnologias e os problemas relacionados com a notação matemática na

web. A proposta que trazemos nesta seção levou em consideração a situação de momento da programação para web. A expectativa é que, com a evolução, os navegadores venham a aderir aos padrões *MathML* e *Unicode*.

Para a função de apresentação dos símbolos e sentenças matemáticas, nossa proposta adota o uso de imagens contendo o desenho das sentenças. Essa escolha justifica-se no fato de que as demais alternativas (*jsMath*, *MathML*, folhas de estilos etc.) não funcionam consistentemente em editores embutidos nas páginas web. A digitação de sentenças matemáticas costuma acontecer associada à digitação de texto simples. Na maioria das ferramentas de *LMS* web atuais, a digitação de texto não-matemático se dá através de editores embutidos nas páginas, que possibilitam formatações especiais, como negrito, itálico etc., através de comandos implementados em javascript. Mesmo nos navegadores em que o *MathML* funciona em *HTML* básico, pode acontecer de as sentenças matemáticas escritas em *MathML* não serem corretamente apresentadas dentro dos editores. Da mesma forma, soluções como o *jsMath*, que desenha a sentença com auxílio do interpretador javascript embutido no navegador, também podem não executar corretamente a apresentação dentro dos editores. Por isso, o uso de imagens ainda é a escolha mais confiável.

No entanto, como discutido na Seção 0, o uso de imagens tem o inconveniente de impossibilitar a edição. Isso foi contornado em nossa proposta, armazenando, no servidor (*LMS*), uma representação da sentença. Cada imagem carrega consigo um atributo que permite ao *LMS* recuperar a representação da sentença que a gerou.

A edição das sentenças, assim como a sua digitação, é feita com o mini-aplicativo *DragMath*, pois um mini-aplicativo tem os recursos de interatividade necessários para a construção do editor, o que a programação no navegador não proporciona. A seqüência típica de edição é a seguinte: o usuário aciona o editor de texto da ferramenta do *LMS* para enviar uma mensagem; ele deseja incluir uma fórmula ao longo do texto e, ainda dentro do editor, aciona um botão referente ao editor matemático; abre-se então uma nova janela em que o *DragMath* está em execução; após a edição da fórmula, o usuário finaliza o *DragMath* e surge uma imagem referente à fórmula editada na posição do cursor. A fórmula é armazenada no servidor em um formato de representação próprio do *DragMath*.

Quando o usuário deseja modificar uma fórmula já existente no texto, que está dentro de um editor em uma ferramenta, basta que ele selecione a imagem e acione o mesmo botão. O *DragMath* é iniciado em uma nova janela já com a fórmula carregada.

Além dessas funcionalidades básicas, é importante que a solução permita interoperabilidade com demais aplicativos matemáticos. Para isso, o usuário pode escolher outros formatos para exportação da sentença, tais como *LaTeX* e *MathML*. Inversamente, deve ser possível importar conteúdos nesses formatos. O *DragMath* já possui o recurso de exportação, mas não o de importação. Nosso trabalho inclui adicionar essas funcionalidades ao mini-aplicativo *DragMath*. Este trabalho está em curso no momento da elaboração deste artigo.

4.1 Trabalhos Relacionados



Esforço semelhante na área de notação matemática em *LMS*, usando tecnologia de código aberto e livre, pode ser encontrado no ambiente Moodle, que utiliza o *DragMath* para geração de códigos *LaTeX* e apresenta a notação matemática usando o *LaTeX* Render. O *DragMath* no *Moodle* não permite alterar uma fórmula criada, a menos que o usuário salve o arquivo no próprio computador. A vantagem da nossa abordagem consiste no armazenamento da fórmula no servidor, assim qualquer usuário que visualize uma fórmula no *LMS*, conseguirá abrir o arquivo para edição e gerar uma nova fórmula.

Outra iniciativa é a *Rwiki* no *LMS Sakai* que apresenta conteúdo digitado em *LaTeX* convertido para notação matemática usando o *jsMath*. Nossa abordagem faz uso do editor *DragMath* para que o usuário consiga de forma mais amigável entrar com conteúdo matemático.

5. Conclusão

Ainda não existe uma solução trivial para escrita de notação matemática na web. Embora padrões tenham surgido, os navegadores, de uma maneira geral, demoram em aderir aos mesmos. Nesse meio tempo, soluções improvisadas surgem constantemente, tentando remediar os problemas.

Apresentamos uma proposta que viabiliza o uso da notação matemática na web tendo tecnologias gratuitas e abertas como base. Essa proposta foi desenvolvida especialmente para o cenário das aplicações web, que contam com servidores de páginas dinâmicas como infra-estrutura.

O trabalho encontra-se no estágio de desenvolvimento. Embora boa parte das funcionalidades já venha pronta desde a origem, algumas adaptações precisam ser feitas, tais como a importação de sentenças em formatos especiais por parte do editor *DragMath*.

À medida que o cenário melhorar, com a adoção consistente de padrões por parte dos fabricantes de navegadores, a solução proposta deverá ser atualizada para contemplar as mudanças. A expectativa é que o padrão *MathML* e *Unicode* ganhem força e aceitação e venham a ser uma solução completa e universal futuramente.

Agradecimentos

O bolsista de treinamento técnico que participou deste trabalho recebeu auxílio financeiro da FAPESP (Processo n. 2005/60417-6).

Referências

BILLINGSLEY, A. & SANGWIN, C. “*DragMath*”, 2008. Disponível em: <http://www.DragMath.bham.ac.uk/>, Março. Acesso em: Set. 2008.



CAPROTTI, O. , CARLISLE D. P. & COHEN A. M. **The OpenMath Standard**, 2000. Disponível em: <http://www.openmath.org/standard/om10/omstd.pdf>, Fevereiro. Acesso em: Set. 2008.

CERVONE, D. P., “**The current state and future of jsMath**”, “**IMA Workshop: The Evolution of Mathematical Communication in the Age of Digital Libraries**”, University of Minnesota, 2006. Disponível em: <http://www.ima.umn.edu/newsletters/2006/12/>, Dezembro. Acesso em: Set. 2008.

FATEMAN, R. J. & CASPI, E., “**Parsing TEX into Mathematics**”, University of California, Berkeley, 2004. Disponível em: http://www.cs.berkeley.edu/~fateman/papers/parsing_tex.pdf, Maio. Acesso em: Set. 2008.

HUTCHINSON, I., “**TtH: a TeX to HTML translator**”, 2007. Disponível em: <http://hutchinson.belmont.ma.us/tth/>, Março. Acesso em: Set. 2008.

KNUTH, D. E., **The TeXbook**, Addison-Wesley, 1984, 15ª edição.

KORPELA, J. (2008) “**Math in HTML (and CSS)**”, 2008. Disponível em: <http://www.cs.tut.fi/~jkorpela/math/>, Junho. Acesso em: Set. 2008.

LAMPORT, L., **LaTeX: A Document Preparation System**, Addison-Wesley, 1994, 2ª edição.

MAYER, S., “**LaTeXRender**”, 2008. Disponível em: <http://www.mayer.dial.pipex.com/tex.htm>, Abril. Acesso em: Set. 2008.

MINER, R., “**The Importance of MathML to Mathematics Communication**”, 2005. Disponível em: <http://www.ams.org/notices/200505/fea-miner.pdf>, Maio. Acesso em: Set. 2008.

SANDHU, P., “**The MathML Handbook**”, Charles River Media, 2002, 1ª edição.

THE *UNICODE* CONSORTIUM, **The Unicode Standard**, Version 5.0, 2006, Addison-Wesley, 5ª edição.

WHITE, S., “**Math in HTML with CSS - Experiments in simple mathematical formatting using common web Technologies**”, 2008. Disponível em: <http://www.zipcon.net/~swhite/docs/math/math.HTML>, Abril. Acesso em: Set. 2008.