

数式の処理と表示のマッシュアップを実現する Maxima 活用システムの開発

Developing a Maxima-based Web System mashing up Algebraic Manipulation and Viewing/Plotting Math Equations

中野 裕司^{*1}, 賈 雲鵬^{*2}, Muhammad Wannous^{*2}, 永井 孝幸^{*1}, 喜多 敏博^{*3}Hiroshi NAKANO^{*1}, Jia YUNPENG^{*2}, Muhammad Wannous^{*2}, Takayuki NAGAI^{*1}, Toshihiro KITA^{*3}熊本大学^{*1} 総合情報基盤センター, ^{*2} 大学院自然科学研究科, ^{*3} eラーニング推進機構^{*1} Center for Multimedia and Information Technologies, Kumamoto University^{*2} Graduate School of Science and Technology, Kumamoto University^{*3} Institute for e-Learning Development, Kumamoto University

Email: nakano@cc.kumamoto-u.ac.jp

あらまし：オープンソースの数式処理ソフトウェアとしてよく知られている Maxima をクロスドメインから利用可能な Web アプリケーションを構築し、LMS 上やローカルなファイルとしてのコンテンツからも数式の処理や表示を可能にするシステムを開発している。本システムを用いることで、eラーニングコンテンツ中で、微積分や展開等数式処理を動的に活用できるだけでなく、MathML による数式表示や、プロット API との組み合わせで、グラフ表示も可能となる。クライアントのみでマッシュアップを実現するための JSONP、jQuery、プロットのための FLOT、HTML5 等の技術を組み合わせることで実装を行っているため、コンテンツ側は JavaScript のみで実装可能である。

キーワード：数式処理, Maxima, マッシュアップ, Flot, HTML5, Canvas, クロスドメイン

1. はじめに

理数系学習コンテンツにおいて、動的な数式処理やその結果のグラフ表示は学習内容の理解を助けると考えられるが、LMS 等の Web 上の学習コンテンツでこれを実現するには様々な問題がある。

数式処理を実現するには、専用の処理ソフトや場合により利用範囲に応じたライセンスが必要で、数式表示には TEX 等で出力したもののイメージ化やプラグイン等が必要で、グラフ表示もプロットツールのイメージ出力や専用アプレット等が必要な場合が多い。また、それらを動的に連携して用いるには、コンテンツ配信サーバ側に専用 Web アプリケーションを開発する等の必要があった。また複数の Web サイトの機能を連携するには、セキュリティ上の理由からのクロスサイトスクリプティングとの衝突を回避しなければならない。

これらの問題の解決には、数式処理ソフトの Web フロントエンド⁽¹⁾や入力プラグイン⁽²⁾等で取り組んだ例が多いが、本研究では、拡張性やコンテンツ配置の自由度を重視し、クライアントの JavaScript のみによる複数サービスのマッシュアップによる解決を試みた。具体的には、オープンソース数式処理ソフト Maxima⁽³⁾, JSONP⁽⁴⁾, Ajax (jQuery), MathML, FLOT⁽⁵⁾ (HTML5 の Canvas⁽⁶⁾利用)を活用により、その実現性と応用可能性を示すことができたので紹介する。

2. システム構成

図 1 に開発したシステム構成を示すが、全ての機能は Web ブラウザから利用できる。数式の処理や表示、さらに数値計算に関して、オープン

ソース数式処理システムの Maxima⁽³⁾をエンジンとし、その機能をクロスドメインでクライアントから呼び出せる WebAPI を JSONP⁽⁴⁾形式で実装した。本 API を用いると、学習コンテンツが他のサーバ (LMS 等) にあっても、またたとえファイルとして読み込んだとしても、インターネット接続さえあれば全ての機能が利用可能となる。

数式処理に関しては、基本機能として、Maxima が解釈可能なテキスト形式で API に渡すことで、結果を Maxima の通常のテキスト出力形式 (display2d:

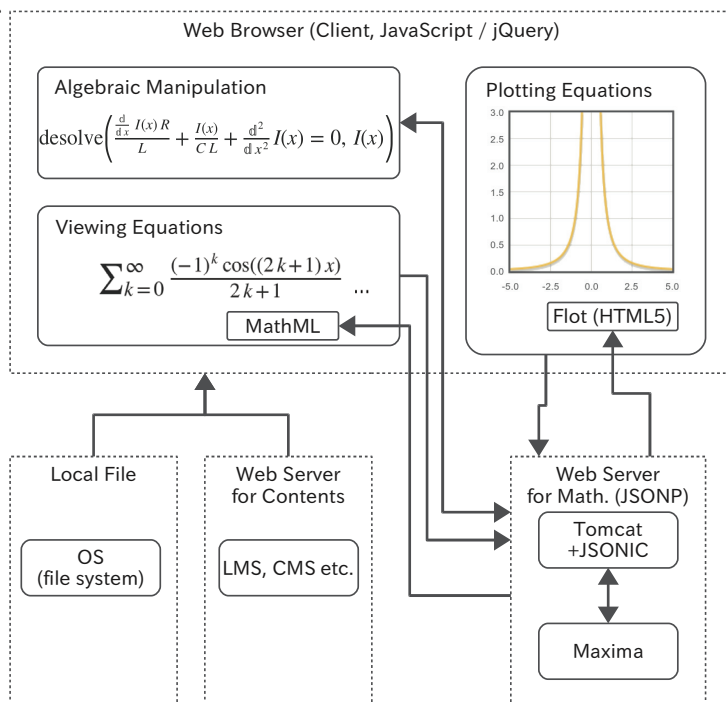


図 1 システム構成

true/false)で得ることができる。この出力形式では解り難いため、次に示す MathML を用いて、入出力式を表示可能であるが、式入力はテキスト形式のみであるため今後 DragMath⁽²⁾等の入力支援の導入を検討したい。

式の表示は、Maxima 用の拡張モジュール mathml.lisp により、Maxima の内部表現を MathML 形式に変換しブラウザで表示可能とした。ただし、関数名、ギリシャ文字、添字、配列等、このモジュールだけでは不十分な部分は JavaScript 中の連想配列で変換しているが、まだ完全に対応しきれていない。

グラフのプロットに関しては、一旦 Maxima の plot2d 関数でサーバ内に gnuplot 用中間ファイルを作成させ、その数値データをクライアントで受け取る。それを Flot⁽⁵⁾に渡すことで、グラフ表示を実現している。Flot は、HTML5 の Canvas⁽⁶⁾タグを用いてグラフィックスを描画するため、Flash や JavaApplet 等のプラグインを必要とせず、JavaScript のみで描画を実現できる。

以上の機能は全て jQuery を用いているため、画面遷移を伴わず、計算が終了した順に式やグラフを自然な形で更新することができる。

3. 学習コンテンツへの試験的実装例

図 2, 3 に、本システムをコンテンツに組み込んだ例を示す。図 2 は静的コンテンツの例であるが、プラグインやイメージファイルを一切使用せず、数式やグラフの表示を実現しており、HTML ソースの変更のみで数式やグラフを変更できることがわかった。

図 3 は動的なコンテンツの例で、数式や表示範囲を入力するとその都度処理を行い、結果やグラフが変更される。図 3(a)は、フーリエ級数の例で、次数等に応じた式の展開とその結果が MathML による式とグラフで表示される。(b)は、微分方程式の解の例で、Maxima により微分方程式を解析的に解き、その解の一部をプロットしている。これらの例に限らず、ユーザの入力した任意の式に関してほぼ同様に動作することを確認した。

まだ、数式の入力方法、MathML への不完全な対応、グラフが 2D のみ等の未実装や課題があるが、提案手法が実現可能であることと、柔軟で幅広い応用の可能性があることを示すことができた。

参考文献

- (1) webMathematic(http://www.wolfram.com/products/webmathematica/), MapleNet(http://www.maplesoft.com/products/maplenet/), Sage (http://www.sagemath.org/), MaximaPHP (http://maximaphp.sourceforge.net/)等
- (2) DragMath (http://www.dragmath.bham.ac.uk/), MathEdit (http://wme.lzu.edu.cn/mathedit/)等
- (3) ホーム: http://maxima.sourceforge.net/
- (4) 高橋登史朗,「jQuery 入門—魔法の JavaScript リファレンス&サンプル」, 秀和システム, (2009/11) p.349.
- (5) ホーム: http://code.google.com/p/flot/
- (6) HTML5 Canvas, http://www.w3.org/TR/html5/the-canvas-element.html#the-canvas-element

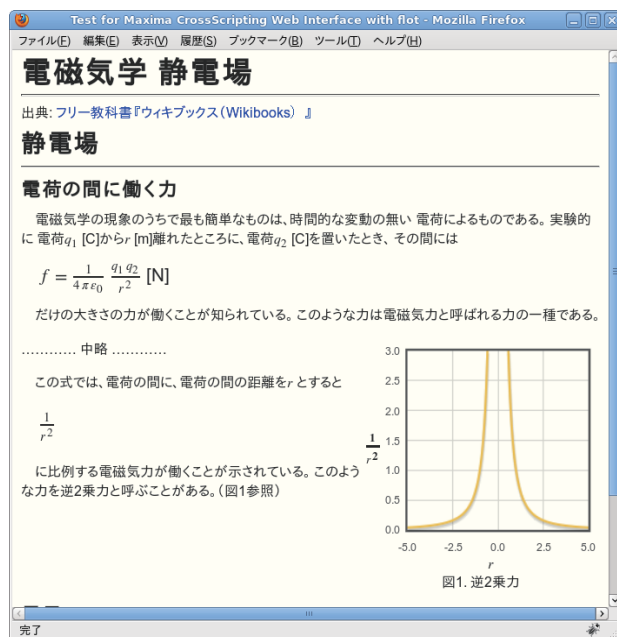
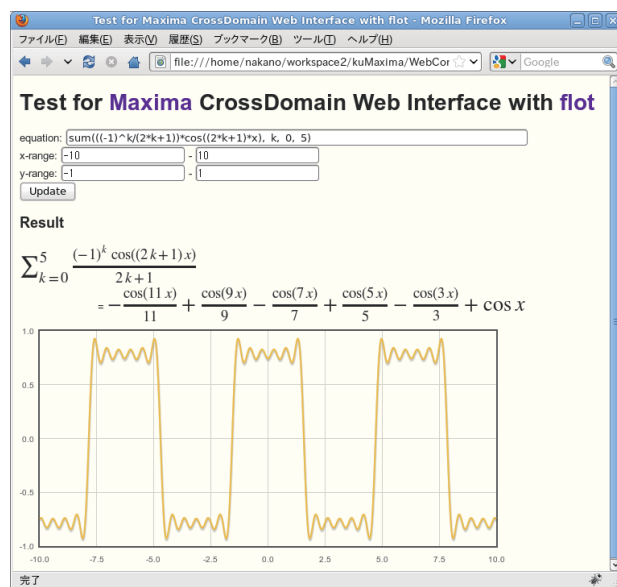
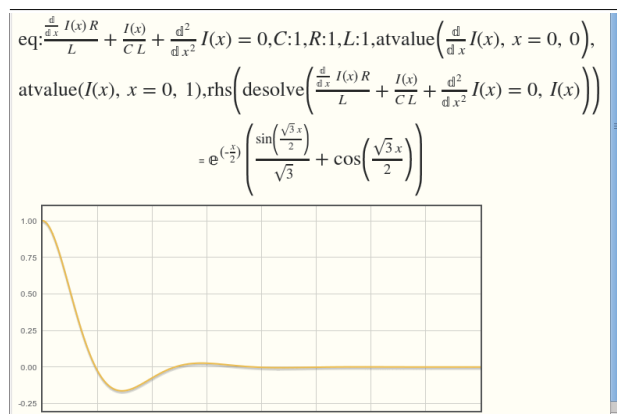


図 2 コンテンツへの応用例



(a) フーリエ級数



(b) 微分方程式の解(LCR 回路)

図 3 動的な数式入力と計算結果のプロット