

インテリジェントな数式ユーザインタフェース (Webアプリケーション版数式エディタの開発)

福井 哲夫^{1,a)}

概要: 仮名漢字変換の気軽さで数式のデジタル入力を平易にする、主に標準のキーボードをユーザインタフェースとする技術を平成 23 年に提案した。ユーザが数式を読むのと同じ、単純な文字を入力すれば容易に数式を構築でき、システムが機械学習型辞書を使ってインテリジェントに候補を算出するので、所望する数式要素を対話的に選択するだけでよく、操作が単純である。評価実験を行った結果、GUI タイプの従来技術に比べ 1.75 倍数式入力のタスク達成時間が速いことが判り、本方式の優位性が示された。本研究では提案技術を応用し、Web アプリケーション版数式エディタ (MathTouchWeb) を開発したので報告する。

An Intelligent User Interface Technology for Easy Formatting Digitalized Mathematical Expressions (A Mathematical Expression Editor on Web-Browser)

FUKUI TETSUO^{1,a)}

Abstract: We have invented an interactive user interface technology for easy formatting digitalized mathematical expressions. This technology includes two features that user inputs is simple similar to the user reading a mathematical expression and user operation is simple because our system calculates all possible candidates corresponding to the mathematical expression using the key dictionary by machine learning. From our result of the performance tests, the mean input time for one mathematical expression with our system has been shown to be 1.75 times as quick as that of current technology. In this paper, we report the mathematical expression editor developed as a Web application by using our technology.

1. はじめに

本研究は数式情報をデジタル化するための数式入力ユーザインタフェースに関するものである。

文部科学省は 2020 年度までに初中等教育における紙の教科書に代わって、デジタル教科書の全面導入を提言している [1]。したがって、数式のデジタル的取り扱いの必要性が高まると予想される。本研究のねらいは、中学・高等学校に導入されようとしている学習者用デジタル教科書の数学科目などにおいて、数式入力を平易にすることによって個別学習を支援することにある。

本発表では、提案方式による数式入力ユーザインタフェースの Web アプリケーション版実装システムの開発と、ユーザビリティ評価について報告する。

2. 新技術の概要

入力された線形文字列から機械学習型辞書を使って、ノードが数式候補であるような決定木を算出し、ユーザとの相互作用によって所望する数式を構築する新方式を 2011 年 8 月に提案した [2]。その後、そのアルゴリズムの定式化と効果・効率についても報告した [3][4]。

2.1 数式文字列表記法

まず初期入力のための数式文字列表記法を次のように定める。

¹ 武庫川女子大学
Mukogawa Women's Univ., Ikebiraki-cho, Nishinomiya,
Hyogo 663-8558, Japan

^{a)} fukui@mukogawa-u.ac.jp

数式文字列表記法

所望する数式を、ユーザが読む順番にその数式要素に対応するキー文字（列）によって区切りなく線形に並べる。

本表記法に従った線形文字列形式の例を表1に示す。例えば、変数 a や α などはいずれもキー“a”で表す。このように、数式要素に対するキー文字（列）はASCIIコードからなり、数式要素を連想しやすい頭文字やLaTeXなどに準じた文字列を採用してもよい。また、演算子の分数記号は“/”，積分記号は“int”などで表し、その前後には分数の分母・分子など、作用範囲の式を表す文字列が並ぶ。ユーザの負担を少なくするために、キーを並べる順番として、人がその数式を読む順番を採用した。特に他の表記法と大きく異なる点は、暗黙積や冪乗演算のように表記されない記号は、線形文字列に含めないところである。例えば、式 a^2 は“a2”と表記する。

2.2 数式入力方式

上記のように、本システムは入力すべき数式文字列が従来方式に比べてとても単純になっている。

その代わりに、所望する数式を構成している数式記号のスタイルや要素間の区切りや各演算子に対する作用範囲などが省略されており、入力された線形文字列形式の情報だけでは2次元の数式表記が一意的に定まらず、完全にフォーマットすることはできない。そこで、本システムはキー辞書データを用いて数式候補を提示し、仮名漢字変換のように、ユーザに不足情報を補うための簡単な指示（次の指示1）～（指示5）のいずれかを）を要求する。

- (指示 1) 候補列の中から次候補を要求、
- (指示 2) 対象要素の現候補を採択、
- (指示 3) (候補が演算子の場合) 作用範囲変更、
- (指示 4) 作用範囲が複数ある場合の変更対象切替、
- (指示 5) 候補を棄却し、文字列のキーを区切り直す。

本数式入力方式によって、ユーザによる入力・操作ミスがなければ、この5つのユーザ指示による手続きから所望する数式を確実に構築できる[4]。そのような数式の構築過程を図1に示す。ユーザは数式構成要素ごとに判断すればよいので、複雑な式であっても迷うことはない。すべての構成要素が確定すれば2次元の表記の構築が完了する[2]。

数式は、最小単位記号自身であるか、演算子とそれが作用する数式の階層構造（演算子構造）をもち、木構造で表現される。そのため、数式構築アルゴリズムは仮名漢字変

表 1 数式文字列の例

所望数式	本方式の入力	LaTeX (従来) の入力
$\alpha = 2$	a=2	$\backslash Y \text{ alpha}=2$
$\frac{1}{a^2+1} \doteq 2$	1/a2+1=2	$\backslash Y \text{ frac}\{1\}\{a \wedge 2+1\} \backslash Y \text{ doteq } 2$
$\int_a^b c = 2$	intabc=2	$\backslash Y \text{ int-}\{a\} \wedge \{b\}c=2$

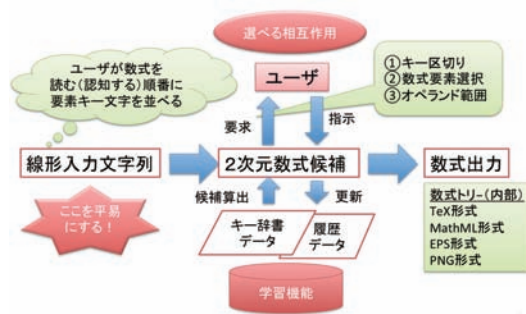


図 1 数式構築過程

換のアルゴリズムと大きく異なる[4]。

2.3 数式入力の実施例

ここでは図1に示した数式構築過程の様子を実例で解説する。

表2に示した擬似コードは、数式 $\frac{1}{a^2+1}$ を構築する過程を、ステップごとの実施画面とユーザの指示コードで表したものである。

まず、ステップ1で、数式文字列“1/a2+1”を入力し、構築処理の開始指示 を打つ。ステップ2では、演算子キー“/”に対する候補と作用範囲がマークされている。次候補要求指示 を押して、所望する分数記号を表示させる。ステップ3で、対象要素採択指示 を押し、次の構成要素キー“a”を対象を移す。ステップ4では、変数 a が候補として強調表示されている。そのまま対象要素採択指示 を押し、次の構成要素に対象を移す。このとき、数式最小単位記号が並ぶ“a”と“2”の間には、省略演算子があると解釈され、ステップ5では、その省略演算子の第1候補である暗黙積演算子*1が選択対象となる。このように、本システムでは数式最小単位記号が並ぶ間を「区切りポイント」と呼んで、特別な処理がなされる。省略演算子は表示されないため対象要素を強調するためのボックスは見えないが、作用範囲はマークされている。このとき

表 2 数式入力の実施例

ステップ	表示画面	指示コード
1:	1/a2+1	<input type="text" value="Space"/>
2:	<input type="text" value="1"/> / <input type="text" value="a2+1"/>	<input type="text" value="Space"/>
3:	<input type="text" value="1"/> / <input type="text" value="a2+1"/>	<input type="text" value="Enter"/>
4:	<input type="text" value="a"/> <input type="text" value="2"/> +1	<input type="text" value="Enter"/>
5:	<input type="text" value="1"/> / <input type="text" value="a2+1"/>	<input type="text" value="Space"/>
6:	<input type="text" value="1"/> / <input type="text" value="a^2+1"/>	<input type="text" value="←"/>
7:	<input type="text" value="1"/> / <input type="text" value="a^2+1"/>	<input type="text" value="Enter"/>
8:	<input type="text" value="1"/> / <input type="text" value="a^2+1"/>	<input type="text" value="Enter"/>
9:	<input type="text" value="1"/> / <input type="text" value="a^2+1"/>	

*1 本システムでは記号の単なる横並びであっても、冪乗関係 \square^\square や添字関係 \square_\square に変換できるため、暗黙の積を一つの演算子として扱う。

範囲変更対象作用範囲は $2+1$ となっている。そこで対象の暗黙積演算子に対して次候補要求指示 $\boxed{\text{Space}}$ を打つと、冪乗演算子に切り替わる (ステップ6)。しかし、指数部の作用範囲が所望する範囲と異なるので、範囲縮小指示 (例えば $\boxed{\leftarrow}$ キー, 拡大する場合は $\boxed{\rightarrow}$ キー) を押して、ステップ7の表示画面のように範囲を変更する。そこで、冪乗演算子 (ステップ7) および次の $+$ 演算子 (ステップ8) に対して、対象要素採択指示 $\boxed{\text{Enter}}$ を押せば、数式の構築が完了する (ステップ9)。

2.4 本技術の Web アプリケーションへの実装

本数式デジタル化技術を応用して、Web アプリケーション版数式エディタ (MathTouchWeb) を Java アプレットにより開発した。操作手順は2.3節で示したように、ユーザー経験を生かすため、仮名漢字変換とできるだけ類似の操作で行えるように配慮した。

実行画面を図2に示す。変換・確定指示の対象である数式要素 (候補記号) が黄色で強調表示されている。図3は指示対象が演算子の場合で、作用範囲 (この例では分数演算子に対する分子, 分母) の式がグレーで網掛けされ、 \leftarrow \rightarrow キーで範囲調整が可能である。

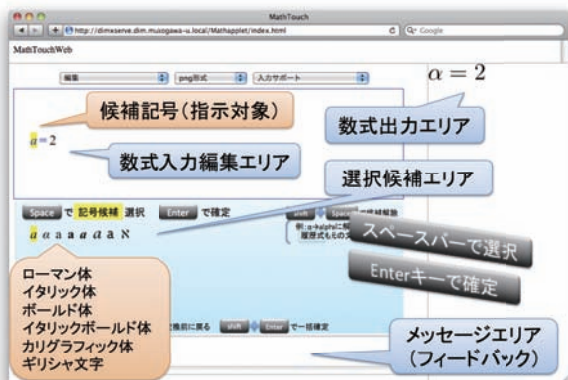


図2 MathTouch 実行画面 (記号選択)



図3 MathTouch 実行画面 (演算子選択)

数式編集機能

扱える数式の範囲は、LaTeX で構築できる数式のほぼ9割をカバーしており、大学の教養数学程度の数式は入力可能である。

しかも、確定した数式に対して、さらに追加または訂正編集も可能となっている [5]。

また、構築した数式は現時点で次の (形式1) ~ (形式4) の出力が可能である。

(形式1) LaTeX 形式

(形式2) MathML 形式

(形式3) PNG 形式

(形式4) JPEG 形式

図2中央の選択メニューで希望の出力形式を指定すると、サーバ側で数式オブジェクトを生成し、クライアント側にそのURLを渡して表示されるので、必要に応じてダウンロードできるようになっている。

3. 従来技術との比較

数式入力のための従来技術として次が代表的である。

従来技術1) 組版指示文字列コンパイル方式

(例: LaTeX, MathML など),

従来技術2) GUI テンプレート方式

(例: 数式エディタ [6] など),

従来技術3) 手書き入力方式

(例: Windows7 の数式入力パネルなど)

1) は組版指示文字列を入力し、整形プログラムによって数式を構築するタイプで、文法が複雑である。2) は数式構造を表すテンプレートをGUIによって選択しながら構築するもので、長い式の編集では煩わしい場合がある。3) の場合、入力は最も自然であるが、誤認識や入力ミスによる編集操作は文書編集に比べ煩わしく感じる。また、入力効率は紙に直接記入する場合と変わらない。

いずれの方式も理数系を専門としない一般ユーザーにとっては依然使い易いとは言いがたい。この理由は、システムが完全に数式構築できる情報をユーザーの操作開始時に要求する仕組みにある。

3.1 コンパイル方式との数式文字列比較

ここでは、従来技術1) の組版指示文字列コンパイル方式と本方式との数式入力文字列の文字数比較を行ったので報告する。

比較対象は表1に示した3つの式に対して、本方式とLaTeXの入力文字数を比較した。その結果を表3に示す。平均で本方式は約6.3文字に対して、LaTeXの場合は約18.3文字となり、約3倍初期の入力文字が多いことが判る。ただ、本方式は数式文字列入力後に要素ごとの候補選択・確定指示を行う必要があるため、表3の3列目に示したように、本方式の総打鍵数はLaTeXとほとんど変わらない。

表 3 数式文字列の文字数比較

所望数式	本方式		LaTeX
	入力文字数	本方式 総打鍵数	入力文字数
$\alpha = 2$	3	8	10
$\frac{1}{a^2+1} \doteq 2$	8	28	28
$\int_a^b c = 2$	8	17	17
平均	6.3	17.7	18.3

ことが判る。このように、本方式は初期入力の負担が軽いという特徴をもつ。

例えば式 $\frac{1}{a^2+1}$ を LaTeX で入力する場合のユーザモデルを考えると、まず行動 (U1)~(U3) をとることになる。

(U1) 全体の式構造を判断して、分数であることを把握、

(U2) 要素の文法判断をして、分数の文法が

$\frac{\text{分子}}{\text{分母}}$ という前置型であることを意識、

(U3) コマンド $\backslash\text{frac}$ を打鍵する。

つづいて、分子および分母部分の入力もほぼ (U1)~(U3) と同様の行動を繰り返す必要がある。LaTeX の表記法は完全に組版できる情報を含むので、入力が終われば構築は完了するが、正しく入力するまでの負担は大きい。

また、従来技術 2) の GUI テンプレート方式も基本的には、式入力時に LaTeX 入力と同じ上記 (U1)~(U3) の繰り返しが必要になることを指摘したい。ただ、(U2) と (U3) の部分が GUI によって分かりやすくなるため心的負担は軽減され、その代わりにマウスによる操作は煩雑になる。

一方、本方式は総打鍵数が同じでも 3分の2 は (指示 1)~(指示 5) のための単純打鍵 (表 2) なので負担が軽い。

3.2 GUI テンプレート方式との効率比較

前節で述べた特徴の違いを裏付けるために、数式入力のタスクテストによって実際に比較実験を行った。

被験者 (武庫川女子大学・情報メディア学科 3 年生) 10 人に同一タスク問題を、本方式実装システム (MathTouch) と Microsoft (登録商標) Word for Mac 2011 ver.14.1.1 に付属する数式エディタ [6] とで実施してもらい、数式入力時間 (タスク達成時間) を測定して比較を行った。タスク問題は次のような分数係数の 2 次多項式である。

$$-\frac{5}{7}x^2 - \frac{5}{8}x + 3, \quad -7\alpha^2 + \frac{9}{7}\alpha - \frac{5}{4}, \quad -\frac{9}{7}\beta^2 - \frac{7}{4}\beta - 4$$

10 問テストを 3 回実施した後の結果を図 4 に示す。縦軸が一つの数式を入力するのに要する平均入力時間 (秒) であり、上側の曲線が数式エディタの結果で、下側の曲線が本システムの結果を表す。t 検定の結果、有意差 ($t(9) = 4.83, p < .01$) があり、このタスクでは、表 4 のように、本システムの方が約 1.75 倍速いことが判った。

表 4 2つの数式入力システム比較

比較項目	習熟曲線 [7]	タスク達成時間 (秒/数式) (3 回目の平均入力時間)
	(R^2 :決定係数)	
数式エディタ	対数線形	34.4 ± 6.2
本システム	対数線形	19.6 ± 2.4

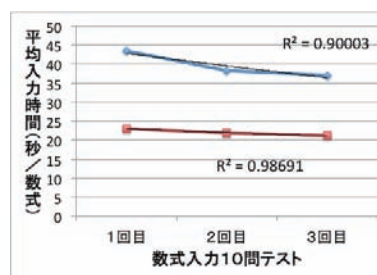


図 4 タスク達成時間の比較 (3 回の習熟曲線)

4. 考察

以上、本提案技術を応用し、Web アプリケーション版数式エディタ MathTouchWeb を開発した。これにより、様々な動作環境で利用が可能となる。

4.1 進歩点 (独創性・優位性)

本提案方式の独創性は、数式に対する文字入力をユーザが読む順番 (認知行動) に対応させた点にある。したがって、従来技術に比べ、

- I. 数式構築のための入力文字列が単純、
- II. 候補を選択すればよいので数式構築操作が容易、
- III. 数式辞書の機械学習機能により入力効率が向上といった優位性がある。

4.2 想定される用途と実用化に向けた課題

本技術は、数式エディタなど、あらゆる数学関連ソフトウェアにおける数式入力インターフェースとして組み込まれることを想定している。特に、数式入力時の負担が従来技術に比べ軽いことから、生徒や学生など一般ユーザへの利用開拓が期待できる。

現在、このようなデジタル教科書と連係してデジタルノート作成や数式の練習問題が可能なデジタル学習ソフトウェアの開発を計画しており、多くの研究者からの技術協力を希望している。

参考文献

- [1] 文部科学省, 教育の情報化ビジョン~21 世紀にふさわしい学びと学校の創造を目指して~, 文部科学省 (2011).
- [2] 福井哲夫:数式のインテリジェントな線形入力方式, 数理解析研究所講義録「数学ソフトウェアと教育」, **1780**,160-171(2012).
- [3] 福井哲夫:数式のインテリジェントな線形入力方式と評価, 数式処理, vol.18 No.2,47-50, 日本数式処理学会 (2012).
- [4] 福井哲夫:線形文字列変換による対話型数式入力方式の効果, 数理解析研究所講義録「数式処理」, **1785**,32-44(2012).
- [5] 福井哲夫:線形文字列変換による機械学習型数式入力インターフェースと編集機能の設計, 情報処理学会第 74 回全国大会論文集,2F-1,4.1-4.2(2012).
- [6] Microsoft(R) co.:マイクロソフト数式エディター 3.0, <http://office.microsoft.com/>.
- [7] 師岡孝次:習熟性工学, 建帛社 (1994).