

# 数学 e ラーニングのための数式予測変換方式による リッチテキストエディタの試作と評価

白井 詩沙香<sup>1,a)</sup> 中原 敬広<sup>2</sup> 福井 哲夫<sup>3,b)</sup>

**概要:** オンライン環境での数式の取り扱いが難しく、理数系 e ラーニングにおいて、学習者のつまずきの要因の一つとなっている。教育の情報化が進捗するなか、これまで手書きで提出していたレポート課題を Learning Management System を通じて電子的に提出する機会も増えており、数式入力への改善は喫緊の課題である。本研究では、理数系 e ラーニングにおける数式混じりの文書入力時の負担改善を目的に、数式予測変換方式によるシームレスなインターフェースを備えるリッチテキストエディタを試作し、評価を行った。その結果、テキストベース入力方式と比べて、1.8 倍速く数学文書を入力ができることがわかった。

**キーワード:** 数学 e ラーニング, 数式入力, LMS, リッチテキストエディタ, 数式文書

## Evaluation of Prototype of Rich Text Editor for Math E-Learning using Predictive Conversion from Equational Text

SHIZUKA SHIRAI<sup>1,a)</sup> TAKAHIRO NAKAHARA<sup>2</sup> TETSUO FUKUI<sup>3,b)</sup>

### 1. はじめに

理数系 e ラーニングの分野では、数学教養科目や入学前教育・リメディアル教育を中心に、オンライン教育が導入されてきた。特に、数式処理システムを活用し、数式で入力された解答の自動採点・評価を行う数学オンラインテストシステムは、国内外の高等教育機関で導入が進められ、利用が拡大している [1]。さらに、2020 年以降は、新型コロナウイルス感染症の拡大防止とともに、学生の学修の機会を保障するために、オンライン教育の導入が急速に進められた。これにより、これまで e ラーニングを導入していなかった授業でも Learning Management System (LMS) の利用が始まり、手書きで提出していた数学のレポート課題を LMS のレポート提出機能を用いて提出させるなど、教育のデジタル化が進められている。

教育のデジタル化の利点として、LMS 等を用いた学習データの蓄積と蓄積したデータを教育改善のために利活用することが挙げられるが、2020 年 9 月に日本学術会議から発表された「教育のデジタル化を踏まえた学習データの利活用に関する提言」[2] では、学習データの利活用を推進するために学習活動に関するデータを収集するための情報環境の整備が必要であることが指摘されている。提言の中では、主に情報端末やインターネット環境を中心に、情報基盤整備の必要性が述べられているが、それと同時に重要となるのが、LMS をはじめとする情報教育システムのユーザビリティの向上である。

理数系 e ラーニングでは、数式入力時の負担が大きく、数式入力インターフェースのユーザビリティが課題となっている。例えば、LMS に搭載されているリッチテキストエディタには、数式入力エディタが搭載されているが、例えば、 $\LaTeX$  形式などマークアップ言語による数式入力エディタは、キーボードで入力が可能であり、習熟している教員・学生にとっては扱いやすく、比較的早く入力できるが、 $\LaTeX$  未経験者にとっては入力時の負担が大きい。一

<sup>1</sup> 大阪大学

<sup>2</sup> 合同会社三玄舎

<sup>3</sup> 武庫川女子大学

a) shirai@ime.cmc.osaka-u.ac.jp

b) fukui@mukogawa-u.ac.jp

方, GUI 形式の数式入力は, 2次元表記の数式構造テンプレートを用いて数式を入力するため, 初学者も扱いやすいと考えられているが, 実際にはマウスを使って数式構造を適切な順序で選びながら入力する手順は煩雑であり, 入力コストが高い. オンライン教育の普及に伴い, 数式を入力する機会が増えるなか, こうした数式入力のユーザビリティに関する課題は学習の阻害やオンライン環境での学習時の負担となる可能性があり, 改善が急務である.

本研究では, 理数系 e ラーニングにおける数式混じりの文書入力時の負担改善を目的に, オープンソースの LMS である Moodle で利用できる数式予測変換方式によるシームレスなインタフェースを備えるリッチテキストエディタを提案する. 本稿では, 有効性を検証する調査として行ったリッチテキストエディタのユーザビリティ評価実験の結果について報告する.

## 2. 関連研究

オンライン環境における数式入力のユーザビリティは, 理数系 e ラーニングの分野では長年の課題となっている. 数式は自然言語とは異なり, 2次元構造を持つため, 情報端末での扱いが難しい. 代表的な数式入力方法として, テキスト形式で数式を入力する方式 (以下, テキストベース入力方式) と GUI による数式構造を示すテンプレートを使って数式を入力する方式 (以下, 構造ベース入力方式) があるが, それぞれに課題がある.

テキストベース入力方式は,  $\text{\LaTeX}$  や MathML などのマークアップ言語や数式処理システムのコマンド形式に従って数式を入力する方式である. 2022 年 2 月時点で Moodle の標準的なリッチテキストエディタとして採用されている Atto editor[3] の数式入力エディタでは, テキストベース入力方式を採用している. テキストベース入力方式は数式の構造や意味をコマンドやタグを用いることで 1次元形式で表現できるため, 熟練者はキーボードのみで入力でき, 構造ベース入力方式と比べてマウスを併用することなく早く入力できる [4]. しかし, テキストベース入力方式で表現された数式は可読性が低く, 初心者には直感的に理解できないといった課題を抱えている [5].

構造ベース入力方式の代表的例としては, Microsoft Office の数式挿入機能や Blackboard のリッチテキストエディタの数式入力エディタとして採用されている WIRIS 社の MathType[6] があげられる. GUI ベースのインタフェースであり, テキストベース入力方式と違い, 2次元形式の数式を確認しながら入力ができるため, 一般的に初心者にも優しい入力方式と考えられている. しかし, 構造ベース入力方式は, 数式構造を理解し, テンプレートを入れ子形式で入力していく必要があり, 実際には初心者にも熟練者にも使い勝手の悪いと指摘されている [7]. 構造ベース入力方式による入力手順も, 紙で数式を書く順序とは異なり, 構

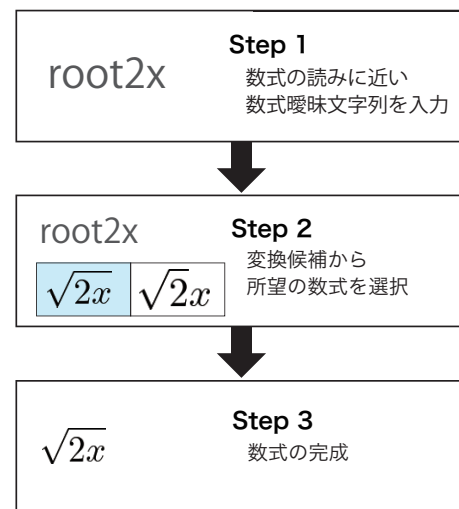


図 1 数式予測変換方式による数式入力

造に合わせて入力の順序を変える能力が求められるため, 直感的ではない [8].

このように, 従来の数式入力方式は見た目や入力順序が紙上で数式を読んだり, 書いたりする際のプロセスと乖離しているため, 直感的な入力ができず, 数式入力時の負担となっている. また, レポート課題のように, 数式混じりの数学文書を扱う際には, 自然言語との連携部分も考慮する必要がある. そこで, 本研究では次章で紹介する数式予測変換方式を用いたリッチテキストエディタを提案する.

## 3. 数式予測変換方式を用いたリッチテキストエディタの試作

### 3.1 数式予測変換方式

数式予測変換方式とは, かな漢字変換のように, 数式の読みに近い数式曖昧文字列を入力し, 変換することで所望の数式が入力できる方式である [9]. 例えば, “ $\sqrt{2x}$ ” と入力するには, テキストベース入力方式である  $\text{\LaTeX}$  形式の場合は “ $\text{\sqrt{2x}}$ ” と入力する必要があるが, 数式予測変換方式の場合は, 図 1 に示すように, 数式構造を指定するための特殊記号は不要で, “root2x” と入力し, 表示される変換候補から所望の数式を選択することで数式入力ができる.

本方式に実装している数式予測アルゴリズムは, 構造化パーセプロトンによる機械学習技術を応用したもので, 数式の各構成要素に出現頻度に応じたスコアを付与し, 上述 “root2x” のような数式曖昧表記文字列から生成可能な全ての数式のうち, 数式を構成する要素のスコアの合計値を算出して, スコアの合計値が最大となる数式を最適な予測候補として提示している [9].

### 3.2 数式予測変換方式を用いたリッチテキストエディタ

前節で述べた数式予測変換方式を実装したリッチテキストエディタ MathTOUCH editor (以下, MT editor) を試作した. 本エディタは Moodle での利用を可能にするため

に、テキストエディタプラグインとして実装している。

MT editor による数式入力の流れを図 2 に示す。提案エディタは、LMS に導入されている一般的なリッチテキストエディタと同じく、フォントやレイアウト等の調整を行う機能を有しており、各ツールは入力エリアの上部に配置したツールパネルから呼び出すことができる。数式入力機能も同じくツールパネルから呼び出す形となり、数式入力ボタン (図 2-①) をクリックすると、カーソルの直下に数式入力エディタが起動し、エディタ上で数式の編集ができる仕組みになっている (図 2-②)。数式曖昧文字列は数式の読みに近いため、直感的に入力することが可能であるが、所望の数式を表す数式曖昧文字列がわからない場合は、数式エディタ上のソフトウェアキーボードに配置されたアイコンから入力できる。そして、数式入力完了後は、ソフトウェアキーボード上の確定キーをクリックするか、物理キーボードの Enter キーを押すことで、完成した数式が挿入される仕組みになっている (図 2-③)。“What You See Is What You Get (WYSIWYG)”で数学文書の編集を進めていけるよう、リッチテキストエディタのテキストエリアには、2次元形式の数式画像が挿入されるようにした。数式画像をダブルクリックすることで数式の再編集も可能である。

## 4. 評価実験

### 4.1 実験概要

提案するリッチテキストエディタの有効性検証として、ユーザビリティ評価実験を行った。評価実験は、20 歳の大学生 11 名を対象に行い、被験者間実験として行った。実験では、実験協力者を 2 群に分け、6 名に MT editor を、5 名に Moodle の標準エディタである Atto editor を使い、微分積分に関する 3 種類の数学文書を入力してもらった。なお、事前アンケートの結果から、いずれの実験協力者も後述するタスク文書で使用した微分積分のテキスト [10] に関する既存知識はないこと、数式入力する機会はほとんどなく、両入力手法ともに未経験であることを確認している。

比較対象となる Atto editor の数式エディタについては、MT Editor と同様にリッチテキストエディタ上のツールパネルに配置されている数式入力ボタンをクリックすることで起動する。数式エディタは  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  を使ったテキストベース入力方式であり、図 3 に示すように、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  形式の文法に従ってテキストを入力する必要がある。代わりに、画面下部に数式プレビューエリアが用意されているため、実験協力者は適宜、プレビュー上で 2 次元形式の数式表示を確認しながら、数式を入力することになる。そして、数式入力完了後は「数式を保存する」ボタンをクリックすることで、テキストエリアに数式が挿入される。なお、Atto editor のテキストエリアには  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  形式の数式文字列が入力される。

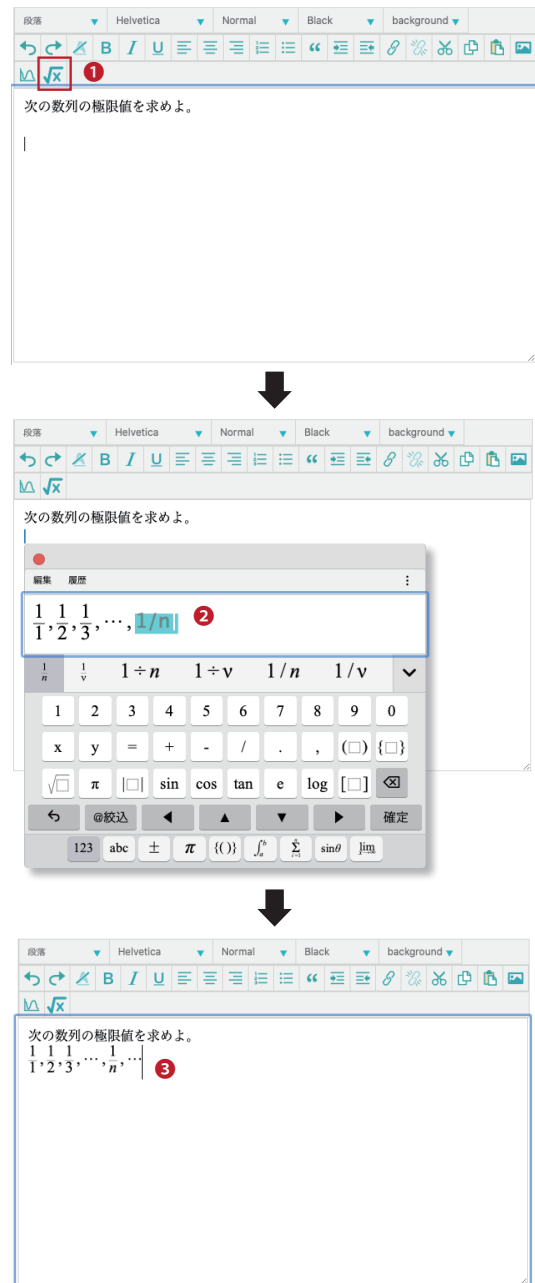


図 2 MathTOUCH Editor による数式入力手順

### 4.2 実験環境および入力タスク

実験では、デスクトップ PC を利用した。モニターサイズは 23.0 インチで、OS は Windows 10 Enterprise 64 bit、CPU は Intel(R) Xeon(R) CPU E3-1230 v5 の一般的な PC である。また、使用した Moodle のバージョンは 3.9.3 で、ブラウザは Google Chrome を利用した。

入力タスクには、文献 [10] に示されている 3 種類の自然言語混じりの数学文書を利用した。表 1 に実際に本番タスクで使用した数学文書を示す。それぞれの文書には 4~5 つの数式が含まれている。また、本番タスクに入る前には、練習として表 2 に示す 10 種類の数式と本番タスクと同質の 3 種類の数学文書を入力してもらった。

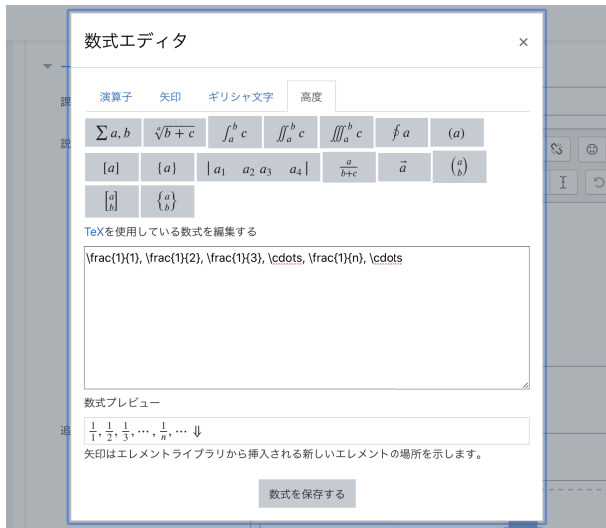


図 3 Atto editor の数式入力画面

表 1 本番タスクで使用した3つの数学文書（文献 [10] より引用し一部改変）

項番	タスク文書問題
問 1	<p>(1) 次の数列の極限值を求めよ。  <math>\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \dots, \frac{1}{2^n}, \dots</math>                      解  <math>2^n &gt; n (n \geq 1)</math> であるから  <math>0 &lt; \frac{1}{2^n} &lt; \frac{1}{n}</math> となり                      例 1.4 より <math>\frac{1}{n} \rightarrow 0</math> であるから  <math>\lim_{n \rightarrow \infty} \frac{1}{2^n} = 0</math>                      がいえる。</p>
問 2	<p>(2) 次の関数 <math>(x^2 + 1)^4</math> を微分せよ。                      解  <math>y = x^2 + 1, z = y^4</math> とおけば  <math>\frac{dy}{dx} = 2x, \frac{dz}{dy} = 4y^3</math> であるから  <math>\frac{dz}{dx} = 2x \cdot 4y^3 = 8x(x^2 + 1)^3</math></p>
問 3	<p>(3) 関数 <math>x^3(x^4 + 1)^8</math> の不定積分を求めよ。                      解  <math>u = x^4 + 1</math> とおく。 <math>du = 4x^3 dx</math> であるから  <math>\int x^3(x^4 + 1)^8 dx = \int u^8 \frac{du}{4}</math>  <math>= \frac{1}{36} u^9 = \frac{1}{36} (x^4 + 1)^9</math></p>

### 4.3 評価指標

定量的評価には、練習時間と3種類の数学文書の入力時間および入力ミス件数を用いる。実験は Moodle の課題機能を使って行い、課題が表示されてから提出されるまでの時間を計測した。入力ミスの件数については、入力途中のミスの定義が困難であるため、提出後の数学文書を対象に、誤入力に含まれる数式の数のカウントした。

メンタルワークロードの評価には、NASA-Task Load Index (以下、NASA-TLX)[11] を利用する。NASA-TLX

表 2 練習で使用した基本数式

項番	基本数式練習問題
1	$x^2 + 5$
2	$\frac{1}{4}$
3	$\frac{1}{4}, \frac{1}{5}, \dots$
4	$a < x \leq b$
5	$\sin^2 x$
6	$4x^3 \cdot 3y^5$
7	$\frac{dy}{du} = 5u^4, \frac{du}{dx} = 2x$
8	$\lim_{x \rightarrow \infty} \frac{1}{x} = 0$
9	$\int x(2x + 5)^3 dx$
10	$= \frac{1}{16} u^5 = \frac{1}{16} (x^2 + 1)^5$

は、主観的なメンタルワークロードを評価するための指標であり、精神的要求、身体的要求、時間的切迫感、作業達成度、努力、不満の6項目について、「低い/高い」「良い/悪い」といったスケールで、項目毎に1~100点で評価を行なってもらい、その平均値を求める [12]。つまり、NASA-TLX の平均値が低いほどメンタルワークロードが低いことを示している。NASA-TLX の平均値の算出方法は様々あるが、本研究では単純平均値 RTLX (Raw Task Load Index) [13] を用いる。

ユーザビリティの評価には、System Usability Scale (以下、SUS) [14] を用いた。SUS はユーザビリティに関する10項目の質問から構成され、「全く同意しない」から「強く同意する」までの5段階のリッカート尺度で回答する。SUS のスコアは、奇数項目・偶数項目毎に処理を行い、各項目を0~4までの数値に変換し、それらの合計値に対して2.5倍することで算出できる。満点は100点であり、ユーザビリティが高いほどスコアが高くなる。

### 4.4 実験手順

実験では、デフォルトのエディタを Atto editor または MT editor のいずれかに設定した個別の Moodle アカウントを用意した。実験協力者には、所定のエディタを使い、はじめに基本練習として10種類の数式と応用練習として本番タスクと同質の3種類の数学文書を入力してもらった。その際、実験協力者には、基本練習問題・応用練習問題の入力方法が書かれたマニュアルを配布し、マニュアルを確認しながら各自で練習タスクに取り組んでもらった。なお、マニュアルには最低限の数式入力方法のみが書かれており、エディタ間で差がないようにした。練習問題終了後は、本番タスクとして表1の3種類の数学文書を入力してもらった。本番タスク終了後には、NASA-TLX, SUS 等のアンケートに回答してもらった。

#### 4.5 結果と考察

練習時間と各本番タスクに要した時間を表3に示す。なお、テキストベース入力方式群の実験協力者1名が全問、もう1名が問1のみ誤った課題を提出しており無効データとなったため、有効データ数は問1は9、問2・3は10となっている。マン・ホイットニーのU検定を用いて検定を行った結果、練習時間並びに全本番タスクにおいて有意な差がみられた ( $p < .05$ )。問1～問3のタスク時間の平均は、Atto editorが8.06分、MT editorが4.40分であり、提案エディタの方が有意に1.8倍近く早く数学文書入力ができていることが判った。MT editorの場合は、数式の読みを入力し、変換することで数式入力ができるが、Atto editorでは、構造を全てマークアップする必要があり、入力文字数が多くなるのが原因と考えられる。さらに、録画した動画を確認したところ、数式プレビューを確認しながら入力を進めることになるため、時間がかかっていることもわかった。また、Atto editor群において、テキストエリアに数式を挿入した後に、2次元形式の数式を確認するために、数式エディタを起動している実験協力者もあり、リッチテキストエリア上で2次元形式で数式が確認できないことも入力時の負担になっている可能性が示唆された。また、MT editor群のSDを確認したところ、Atto editorに比べ、MT editorの値が低く、ばらつきが小さいことから、どの実験協力者も短時間で習熟し、安定して早く入力できていると考えられる。一方、ミス件数については、Atto editor群は0件、MT editor群では2名の実験協力者において、それぞれ4件、1件のミスが発生していた。ミスが発生していた数式の事例を表4に示す。大半のミスが数式の視認性（例えば上付きかどうかの紛らわしさ）に起因するもので、予測変換・候補選択操作で起こりやすいミスであることがわかった。この誤入力は予測変換の精度向上とともに、変換候補の表示領域を広げる等、視認性を向上させることが求められる。

NASA-TLXの結果は、Atto editorが55.4 ( $SD=7.58$ )、MT editorが48.3 ( $SD=5.00$ )となり、2群の間に有意な差はみられなかった。両群ともに50前後の総合値となり、メンタルワークロードの値が若干高いことがわかった。

一方、SUSはAtto editorが71.88 ( $SD=3.70$ )、MT editorが74.58 ( $SD=7.13$ )となり、ユーザビリティに関する主観評価は両群ともに高い評価となった。SUSの結果についてもマン・ホイットニーのU検定を用いて検定を行った結果、有意な差はみられなかった。

#### 5. まとめと今後の課題

本研究では、理数系eラーニングにおける数式混じりの文書入力時の負担改善を目的に、オープンソースのLMSであるMoodleで利用できる数式予測変換方式によるシームレスなインタフェースを備えるリッチテキストエディタ

表3 平均入力時間(分)の結果

	Atto editor		MT editor	
	入力時間	SD	入力時間	SD
練習時間	38.95	5.47	27.27	3.96
問1	9.41	2.07	5.86	0.98
問2	6.40	2.91	3.20	0.80
問3	8.70	2.87	4.13	0.77

表4 誤入力が発生していた数式の例

正しい数式	誤入力例
$\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \dots, \frac{1}{2^n}, \dots$	$\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \dots, \frac{1}{2^n}, \dots$
$x^3(x^4 + 1)^8$	$x^3(x^4 + 1)$
$\frac{1}{36}u^9$	$\frac{1}{36}u^9$
$\frac{1}{36}(x^4 + 1)^9$	$\frac{1}{36}(x^4 + 1)$

を試作した。有効性調査としてユーザビリティ評価実験を行った結果、提案エディタの方が、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 形式のテキスト入力インタフェースをベースとしたリッチテキストエディタより有意に早く数学文書を入力できることが確認できた。一方で、変換候補の視認性の問題も明らかとなったため、UIデザインを調整する予定である。また、今回の評価実験は少人数であったため、今後検証精度を高めるに十分な被験者を募り、数式入力インタフェースの熟練者も対象に、パフォーマンス測定実験を行い、提案インタフェースの効果を検証したい。

#### 謝辞

本研究はJSPS科研費19H01733, 20K12117, 21H04412の助成を受けたものです。

#### 参考文献

- [1] Nakamura, Y., Yoshitomi, K., Kawazoe, M., Fukui, T., Shirai, S., Nakahara, T., Kato, K. and Taniguchi, T.: *Effective Use of Math E-Learning with Questions Specification*, pp. 133-148, Springer International Publishing (2018).
- [2] 日本学術会議: 教育のデジタル化を踏まえた学習データの利活用に関する提言-エビデンスに基づく教育に向けて-(2020).
- [3] Atto editor: Atto editor (online). 入手先 ([https://docs.moodle.org/311/en/Atto\\_editor](https://docs.moodle.org/311/en/Atto_editor)) (参照 2022.02.14).
- [4] 白井詩沙香, 仲村裕子, 福井哲夫: 数式自動採点システムにおける数式入力インタフェースの提案と評価, 情報処理学会論文誌教育とコンピュータ (TCE), Vol. 1, No. 3, pp. 11-21 (2015).
- [5] Pollanen, M., Hooper, J., Cater, B. and Kang, S.: A tablet-compatible web-interface for mathematical collaboration, *International Congress on Mathematical Software*, Springer, pp. 614-620 (2014).
- [6] WIRIS: MathType (online). 入手先 (<https://www.wiris.com/en/mathtype/>) (参照 2022.02.14).
- [7] Padovani, L. and Solmi, R.: An investigation on the dynamics of direct-manipulation editors for mathematics,

- International Conference on Mathematical Knowledge Management*, Springer, pp. 302–316 (2004).
- [8] Pollanen, M., Wisniewski, T. and Yu, X.: Xpress: a novice interface for the real-time communication of mathematical expressions, *Proceedings of MathUI*, Vol. 8, Citeseer (2007).
- [9] Fukui, T. and Shirai, S.: Predictive algorithm for converting linear strings to general mathematical formulae, *International Conference on Human Interface and the Management of Information*, Springer, pp. 15–28 (2017).
- [10] 熊原啓作: 入門微分積分学 15 章, 株式会社 日本評論社 (2016).
- [11] Hart, S. G. and Staveland, L. E.: Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research, *Advances in psychology*, Vol. 52, Elsevier, pp. 139–183 (1988).
- [12] 三宅晋司: 特集 人間工学のための計測手法 第 3 部: 心理計測と解析 (6) メンタルワークロードの計測と解析-NASA-TLX 再考, *人間工学*, Vol. 51, No. 6, pp. 391–398 (2015).
- [13] Byers, J. C.: Traditional and raw task load index (TLX) correlations: are paired comparisons necessary?, *Advances in Industrial Ergonomics and Safety 1: Taylor and Francis*. (1989).
- [14] BROOKE, J.: SUS : A Quick and Dirty Usability Scale, *Usability Evaluation in Industry*, pp. 189–194 (online), available from (<https://ci.nii.ac.jp/naid/10024892921/>) (1996).