

# KeTLTS への Algebrite の組み込みと利用

弓削商船高等専門学校 久保 康幸

Yasuyuki, Kubo, National Institute of Technology, Yuge College

福島工業高等専門学校 西浦 孝治

Koji Nishiura, National Institute of Technology, Fukushima College

KeTCindy センター 高遠 節夫

Setsuo Takato, KeTCindy Center

## 1 はじめに

2020年のCovid-19により授業が大きく変わり、多くの学校でオンライン授業が取り入れられた。数学の授業も例外ではなく、教員たちは、数式をどのようにやりとりするかという問題に直面した。教員から学生へ数式を見せるには、TeXを利用することができる。しかし、学生から教員に数式を送る場合には、多くの学生がTeXを使っていないため、工夫が必要である。画像などを送る場合には、多数の学生から容量の大きい画像を受け取るのは教員の負荷が重いなどの問題がある(図1)。その対応の一つとして、高遠節夫は、2021年に、テキストをベースとした数式送受を目的として、KeTMathルール<sup>1</sup>とKeTLTS (KeTCindy Learning data Transfer System)を開発した([4])。

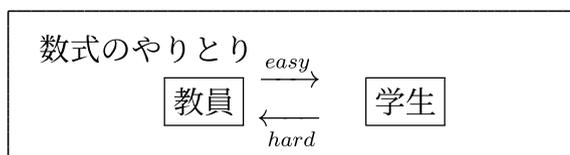


図1：数式のやりとり

KeTMathルールは、2次元の数式を1次元で入力するための規則であり、次のような特徴を持つ。

- (1) CindyJSは、KaTeX (v0.8)を使う。
- (2) KeTMathルールで入力された式をTeX形式にして、HTML上に2次元で表示する。
- (3) KeTMathルールで入力された式を必要に応じて変換する関数として、次を用意している。

Totexform：TeX形式に変換

Tocindyform：CindyScript形式に変換

Tomaxform：Maxima形式に変換<sup>2</sup>

<sup>1</sup>KeTMathルールは、例をあげてホームページで紹介し、普及に努めている([1],[2])。

<sup>2</sup>Algebrite形式の式を得るためにMaxima形式へ変換する関数Tomaxformを転用している。

KeTMath ルールは、実際の授業で学生に利用させ、より習得しやすく使いやすいものとなるように修正が行われている。2025年に、三角比や根号、対数について  $\sin(^{(2)}x)$ ,  $\log(_{(a),b)}$  などの入力規則の追加が行われた。例えば、 $\sin^2 x$  という表示を得るためには、これまで  $\sin(2,x)$  と入力していた (図2の1つ目)。それを累乗のための  $^{(2)}$  を使う学生がいるため、 $\sin(^{(2)}x)$  と入力すると、確認用の窓には、 $\sin(2,x)$  が表示され、2次元表示の式は、 $\sin^2 x$  を得るようにルールを追加した (図2の2つ目)。

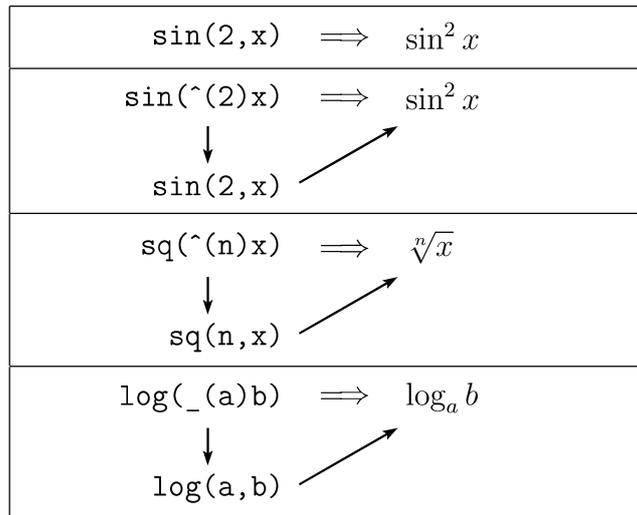


図2：追加されたルールの表示までの流れ

KeTLTSは、テキストをベースとしたLMSの利用を考えて、2021年に発表されたシステムを次第に改善していったものである ([5])。KeTCindyのホームページ ([1]) を参照すれば使えるようになっている<sup>3</sup>。簡単に使い方を説明すると、次のようになる。

- (1) ‘toolketmath.cdy’ を使って、template ファイルに、question データを加えることで ‘kettask(+ID).html’ を作る。question データの質問と正解は、KeTMath ルールで入力する。
- (2) ‘kettask~.html’ (kettask) を例えば Github にアップロードし、その URL を Google Classroom (GC) や他の LMS で学生に配布する。URL は1行で構成され、kettask のサイズは小さく 100~200KB 程度である。また、授業中の適切なタイミングで送信または表示する。

KeTLTSによる課題は、これまで次のような手順で回答の回収と採点をしていた。ここでは、GCの例で紹介する。

- (1) ‘anssheetall.txt’ にメールで送られた回答をコピーする。
- (2) ‘toolketmath.cdy’ の [Anschart] ボタンにより、回答の一覧表(図3)である ‘anschart.csv’ を作成する<sup>4</sup>。

<sup>3</sup> “KeTCindy Home” で検索すれば得られる。

<sup>4</sup> ‘anschart.csv’ はテキストファイルであるから、いろいろな方法で加工できる。



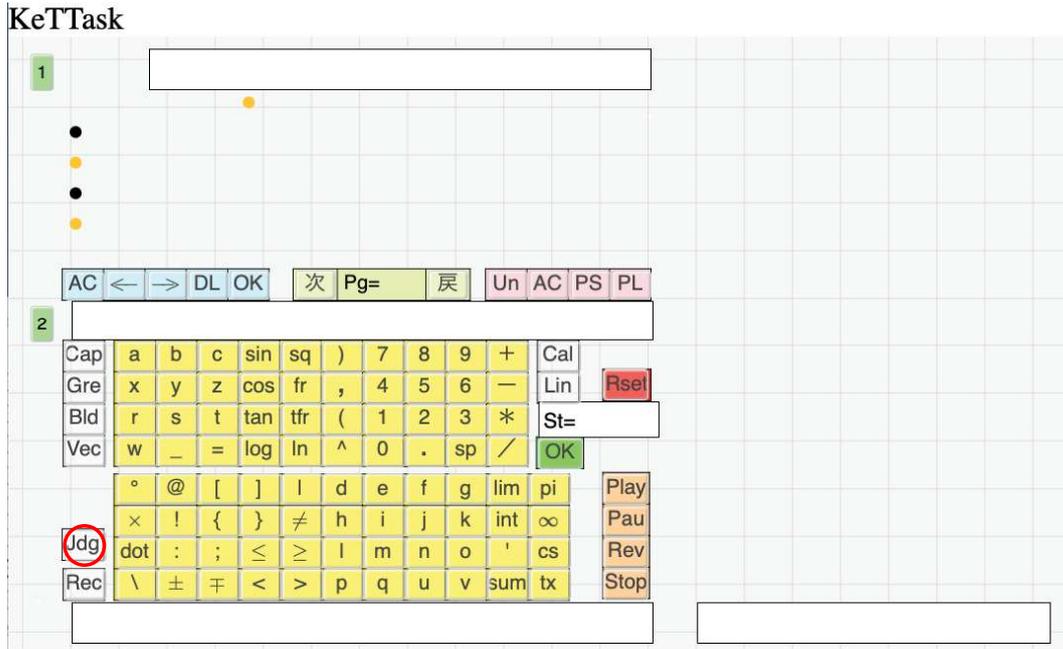


図4：kettaskの始めの画面

## 2.2 kettask の作り方

Algebriteを組み込んで **Jdg** ボタンを追加した kettask は、これまでの KeTLTS と同様の手順で作ることにした。すなわち、Algebrite の組込みと **Jdg** ボタンのありなしを特別な手順として追加しなかった。kettask の作り方は、次のようになる。

- (1) 'toolketmath.cdy' を起動する。
- (2) '1.taskline を作成' と '実行' を順にクリックする (図5)。



図5：手順(2)を実行した後の画面

- (3) '2.Kettaskに組込' をクリックし、一番上のファイルを選んで '実行' をクリックする (図6)。



図6：手順(3)を実行した後の画面

## 2.3 Jdg ボタンによる回答の判定

次に **Jdg** ボタンによる回答の判定の様子を手順を追って紹介する。

- (1) 回答を入力する。
- (2) **Jdg** をクリック（下の窓に‘0’が表示されれば正解）。（図7）

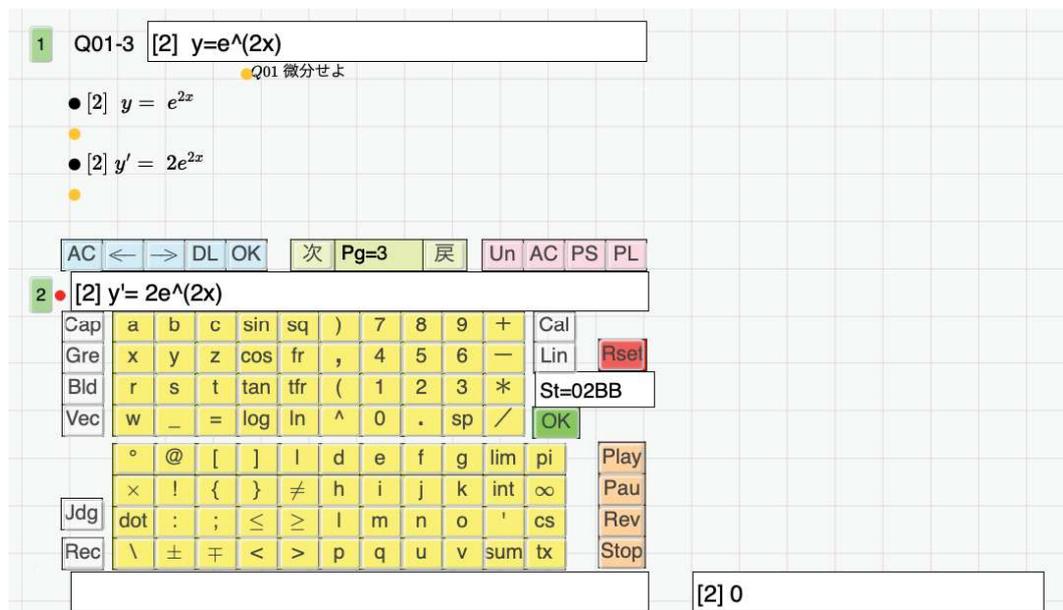


図7：正解により下の窓に‘0’が表示された画面

- (3) **次** をクリック（下の窓の表示が消える）。
- (4) 回答を入力して **Jdg** をクリックすれば、その設問の判定が下の窓に出る。（判定がエラーの場合もある）
- (5) 回答を訂正して **Jdg** をクリックすれば、再判定する。

## 2.4 Algebrite の使うスクリプト

“ $\frac{1}{3}x^3$  を微分せよ ” という設問を例にして、Algebrite の使い方を示す。次の例は、`exealg` を実行するスクリプトから抜き出したものである。

例:

```
tmqu1="fr(1,3)x^(3)";// KetMath ルールで記述
tmqu2=Tomaxform(tmqu1);// Tomaxform により式を Algebrite 形式に変換
tmqu3="d("+tmqu2+",x)";// d(f,x) は、Algebrite の関数
out1 = exealg(tmqu3);
```

この例では、設問の式  $\frac{1}{3}x^3$  を KeTMath ルールで記述した ‘`fr(1,3)x^(3)`’ を `tmqu1` に代入して、さらに Algebrite 形式に直してから `tmqu2` に代入している。正解の式は、Algebrite で微分して得た  $x^2$  を表す式 `x^2` が `out1` に代入される。

## 2.5 Algebrite による判定の特徴

次に、“ $x^3 - 12x$  を微分せよ ” という設問を例にして、Algebrite を利用した正誤判定が文字列の一致でないことを示す。そのスクリプトを抜き出すと、下のようになる。この例では、正解となる式をそのまま `tmqu3` に入力するのではなく、Algebrite に微分させている。また、‘`3x^(2)-12`’ と ‘`3(x^(2)-4)`’ は、どちらも正解であり、Algebrite を利用した判定でも正解と判定される。

例:

```
tans3=Tomaxform(tans2);// tans2 は、学生の回答
tmqu3="d("+tmqu2+",x)";
jdgfun=tmqu3+"-("+tans3+)";
jdgout = exealg(jdgfun);
```

この例で、学生の回答は、`kettask` のシステム内部の変数 `quL`, `nqu` などの組合せによって得ている。

## 3 まとめと今後の課題

今回、KeTLTS で配布する `kettask` に Algebrite を組み込みこと、および、Jdg ボタンの追加により、以下のような効果が得られる。

- (1) 学生は、自身の回答を Algebrite により、回答の送信前に判定することができる。そこで、学生自身が送信前に回答をチェックすることで、提出回答における単純ミス

の減少と教員から助言が減少することが期待できる。さらに、KeTLTSを独習教材としても使うことができる。

- (2) 教材と同じブラウザ上で判定できる。そのことにより、教材との一体感が出る上、学生の行動もコントロールしやすくなる。なお、ブラウザ上で稼働する数式処理システムとして他に Wolfram Cloud などがあるが、その場合、別に立ち上げたブラウザ上で実行することになる。
- (3) Algebrite による採点結果を反映した回答の一覧表 ‘anschart.csv’ (図 8) があれば、図 3 のような表のすべての点数欄に手動で入力したり、Maxima を利用して採点するより、教員の負荷が軽減する。例えば、点数が先にあれば、誤答について部分点を考えればよい。

	A	B	C	D	E	F	G	H	I
1	CA	2025	711		Q01---	[1]4x^3	4 [2]-x^(-2)		4
2	1 01AA		711	13:20:29	Q01---	[1] y'= 4x^(3)	4 [2] y'= -x^(-2)		4
3	2 02BB				Q01---	[1]na	0 [2]na		0
4	3 03CC		711	13:14:16	Q01---	[1] y'= 4x^(3)	4 [2] y'= -fr(1,x^(2))		4
5	4 04DD		711	13:15:39	Q01---	[1] y'= 4x^(3)	4 [2] y'= -2		0
6	5 05EE				Q01---	[1]na	0 [2]na		0

↑  
点数

↑  
点数

図 8：採点後の ‘anschart.csv’ を表計算ソフトで開いた画面

いっぽうで、Algebrite の組み込みと Jdg ボタンを追加したシステムを使ってみて、以下のような課題が考えられる。

- (1) 現在は、KeTMath ルールで入力された式を Algebrite で扱うとき、Tomaxform 関数を転用して、Maxima 形式に変換している。複雑な式に対応できるよう、Algebrite 形式に変換する関数を用意する必要がある。
- (2) 今回、提出前の回答を学生自身が正誤判定することができるようにした。回収された学生の回答の一覧表を教員が採点する場面でも Algebrite を利用できるようにシステムを拡張する。
- (3) Algebrite を使った判定内容を設問の内容に応じて柔軟に対応できるよう改善する。今回の例では、微分の問題であるため、式を整理して一致すれば正解であると判定させた。しかし、積分の問題であれば、積分定数の差を考慮する必要がある。また、因数分解させる問題では、他の判定法を考える必要がある。CindyScript による処理と Algebrite による計算を組み合わせることで、そういった判定に対応する。
- (4) kettask 作成時に、判定内容を html に埋め込む方法を改善して、KeTLTS による教材作成者の負担を軽減する。教材作成者が求める判定内容を実現するのに、Algebrite を深く理解しないと実現できないシステムにしてしまうと、いろいろなユーザーに kettask を使ってもらおう上でのハードルが高くなる。Algebrite を使うには、ある程

度の理解は必要となるが、高いプログラミング能力を要求しないシステムとすることを指す。

- (5) 学生が  Jdg ボタンにより自身の回答を判定できることで、どのような教育効果があるのかを調査する。

## 参考文献

- [1] KeTCindy Home: <https://s-takato.github.io/ketcindyorg/indexj.html>
- [2] Kubo KeTCindy: <https://kubo-yuge.github.io>
- [3] 高遠節夫: KeTCindyJS の開発と教育利用, 京都大学数理解析研究所講究録 2142, pp.123–132, 2019
- [4] 高遠節夫, 濱口直樹: Web 利用の理数教育に役立つ数式送受システムの開発, 京都大学数理解析研究所講究録 2178, pp.67–76, 2021
- [5] 濱口直樹, 北本卓也, 高遠節夫: テキストをベースとした LMS の利用と HTML 教材の作成, 京都大学数理解析研究所講究録 2208, pp.58–67, 2021
- [6] 高遠節夫, 西浦孝治, 牧下英世: ブレンド型授業における KeTLMS の利用, 京都大学数理解析研究所講究録 2301, pp.188–197, 2025
- [7] 北本卓也: Javascript を活用した教材作成, 京都大学数理解析研究所講究録 2142, pp.154–162, 2019
- [8] 北本卓也: HTML5 を用いた授業支援ツールについて, 京都大学数理解析研究所講究録 2178, pp.39–47, 2021
- [9] 北本卓也: 既存教材を活用した E-Learning システム構築のフレームワーク, 京都大学数理解析研究所講究録 2280, pp.70–77, 2024
- [10] 金子真隆, 北本卓也, 野田健夫: オンラインでの動的な探究学習における数学ソフトウェアの活用について, 数理処理 bulletin of JSSAC vol.28, No.2, pp.5–19, 2022