

# Handwriting Interface for Computer Algebra Systems

H. Okamura, T. Kanahori, M. Suzuki

Graduate School of Mathematics, **Kyushu University**  
suzuki@math.kyushu-u.ac.jp

R. Fukuda

Faculty of Engineering, **Oita University**  
rfukuda@cc.oita-u.ac.jp

W. Cong, F. Tamari

Faculty of Education, **Fukuoka University of Education**  
tamari@fukuoka-edu.ac.jp

## Abstract

This paper describes our experimental real time recognition system of handwritten mathematical expressions, linked to a computer algebra system Mathematica by Mathlink. This system is titled as “INFTY”. The fundamental operations of INFTY are (1) input of mathematical expressions using handwriting input system, (2) editing of them by basic editing operation (cut, copy, paste and delete) and (3) calculations and plotting graphs using Mathematica. INFTY also supports the input and output by TeX format.

In the handwriting window, we adopt the automatic character rewriting method. This method largely improves the accuracy of the the structure analysis of the written mathematical expressions.

## 1 Introduction

The recent developments in network and computer technology is allowing us the prospect of adopting new approaches to both the research and the education of mathematics. For example, an electric white board connected

by network to a computer algebra system or some mathematical data base enables us to deliver a new style of lecture or discussion on mathematics.

However, it is true that the user interfaces of the current computer systems are not convenient to input mathematical expressions ([1][2]). There are two typical methods to input mathematical expressions into a computer. The first one is based on menu operations. The user selects mathematical symbols from panels and change modes to input subscripts, superscripts, etc. This method is intuitive and easy to understand for the user, but repetitious menu operations sometimes prevent the smooth input of mathematical expressions and can disturb the train of thought of the user. The second method uses only the keyboard to input mathematical expressions using syntax based data formats, such as TeX notations or command formulations of computer algebra systems, etc. This latter method often results in a smoother input of mathematical expressions. However, some training is required to master the notations and it is not easy to understand the meaning of the written expressions at a glance.

To solve this problem, we are developing a tentative real time recognition system of handwritten mathematical expressions([3][4]). In order to test the efficiency of our system, we linked it to the computer algebra system "Mathematica" by Mathlink. This experimental system is named "Infty".

The fundamental operations of Infty are the followings. First, users write mathematical expressions in a dialog box by hand using a mouse or a data tablet. Each handwritten mathematical expression is recognized and put into the main board (display area) of Infty, transformed into the type character form. The mathematical expressions written on the display area can be edited by the usual basic editing operations such as cut, copy, paste and delete, and they can be evaluated, factorized, expanded or presented by graph, etc. using Mathematica linked by Mathlink. The results of the calculation are put into the clipping board of Infty, and can be pasted at any place of the display area of Infty and edited freely. Infty also supports the input and output by TeX format texts and can be used, for example, to edit mathematical expressions of TeX source by handwriting, making use of calculation results by Mathematica as the need arises.

In the handwriting window, each character is recognized at each step and automatically rewritten in a regular form. This automatic rewriting method considerably improves the accuracy of the structure analysis of the written mathematical expressions.

Presently, Infty recognizes all mathematical expressions, except for matrices, at the high school level. The recognition of complicated expressions at university level is left to future researche.

As our system is developed using object oriented programing designs,

it can be easily extended to support inputs and outputs using any other formats of mathematical expressions, such as Maple format, MathML format or Braille codes, etc. by overriding some primitive functions.

## 2 Infty System

Infty is a handwriting interface for Mathematica, and by using this system we can easily execute calculations or plot graphs. We are developing this system using Visual C++ compiler system (ver. 6.0). Therefore it can be used in Windows 95/98 systems in which Mathematica is installed.

Its main window consists of two areas: Display Area and Handwriting Area. Mathematical expressions for the executions are inputted to Display Area, using a handwriting input system in Handwriting Area. Expressions in Display Area are displayed in type setting form. The calculation results are also displayed in Display Area and these are also objects for the executions. Details for Handwriting Area are described in Section 3. We explain the remainder parts of this system in this section.

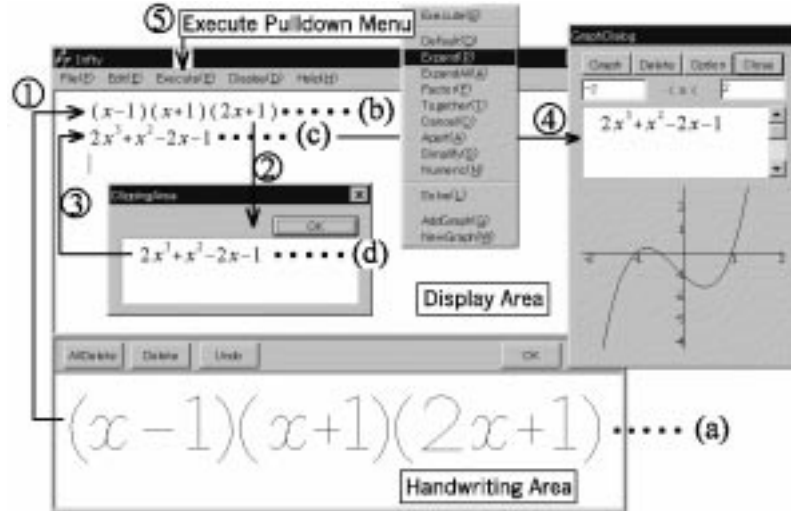


Figure 1: Infty

### 2.1 Display Area and Clipping Area

In Display Area, we can edit mathematical expressions using ordinary editing operations: cut, copy, paste and delete. To select a mathematical expression,

we surround the corresponding expression with the rectangle which is displayed when the mouse is dragged. We may also click the left edge of a line to select the whole expression in the line. When the function cut or copy is executed, selected expressions are stored in the buffer of Infty, and displayed in the dialog box called 'Clipping Area'. The Clipping Area gets visible when the first expression is stored in the buffer. The Clipping Area is also used to represent calculation results by Mathematica. Mathematical expressions in Clipping Area, which are stored in the buffer, can be pasted to anywhere in Display Area.

## 2.2 Execution

Let us explain the menu items in the pull down menu 'Execute' following the Figure 1. First, the expression is input by hand in Handwriting Area (a) Pushing the button (OK), the recognized expression is displayed in Display Area ①. Suppose that the user selected whole of the first line. Next, selecting 'Expand' in the pull down menu 'Execute' (see ⑤ in Figure 1), the calculation result by Mathematica is displayed in Clipping Area (②). If the user wants to continue the same operation 'Expand', the user may select 'Execute' button in the pull down menu or double click the left button on the expression in Display Area.

Each menu item in this pull down menu, except for 'Execute', 'Add Graph' and 'New Graph', directly stands for the corresponding Mathematica command. Figure 2 shows the results of the execution noted below "→" for the expressions on the left hand side written in Display Area.

$\int \frac{1}{x^2-1} dx$	→	$\frac{\log(-1+x)}{2} - \frac{\log(1+x)}{2}$
$\int_0^{\pi} x^2 \sin(x) dx$	→	$-4 + \pi^2$
$\lim_{n \rightarrow \infty} (\sqrt{n^2+1} - n)$	→	$0$
$\sum_{k=1}^n (2k-1)^2$	→	$\frac{n(-1+2n)(1+2n)}{3}$
$x^3+y^3+z^3-3xyz$	→	$(x+y+z)(x^2-xy+y^2-xz-yz+z^2)$
$\lim_{n \rightarrow \infty} (1+\frac{1}{n})^n$	→	$2.71828$

Figure 2: Examples of Execute

## 2.3 Graphic Display Area

Suppose that the expression “ $2x^3 + x^2 - 1$ ” ((c) in Figure 1) is selected. Executing ‘Add Graph’ or ‘New Graph’, the graph of “ $y = 2x^3 + x^2 - 1$ ” is plotted in Graphic Display Area, which is a dialog box for graphics(④ in Figure 1). We can change the range of the variable  $x$ , rewriting two numerals in both side of ‘ $< x <$ ’ in the upper side of Graphic Display Area (see the left dialog box in Figure 1).

If there are already some graphs in Graphic Display Area, we use ‘New Graph’ and ‘Add Graph’ as follows. When ‘New Graph’ is selected, the old graphs are cleared and only the new graph is plotted in the area. On the other hand, when ‘Add Graph’ is selected, the old ones remain plotted and a new graph is plotted on the same area. In both cases, the expression of the function will be listed in Graphic Display Area. In general, there are several functions in this list. We can select some of them by clicking on the corresponding expressions, and remove them by ‘Delete’ button.

Graphic Display Area is not visible at first. Using the menu item ‘Graphics Visible/Invisible’ in the pull down menu ‘Edit’, we can make this dialog box visible or invisible.

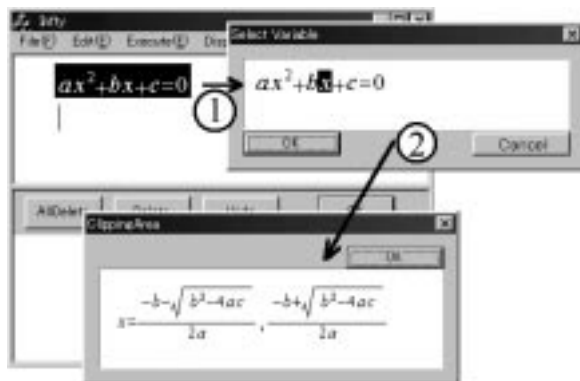


Figure 3: Dialog box for Solve

## 2.4 Solve

Let us explain the execution ‘Solve’ following Figure 3. Assume that the expression “ $ax^2 + bx + c = 0$ ” is input and selected in Display Area. ‘Solve’ is also in the pull down menu ‘Execute’. Followed by the click of ‘Solve’, a dialog box appears to determine the variable to solve the equation for (see ① in Figure 3). We can select the variable by clicking on the character of the

variable in the dialog box. Clicking ⟨OK⟩ button, we can obtain the solution in Clipping Area (see ② in Figure 3).

### 3 Handwriting Interface

User can input mathematical expressions by hand using a mouse or a data tablet in the lower window of the INFITY system. We call this window Handwriting Area.

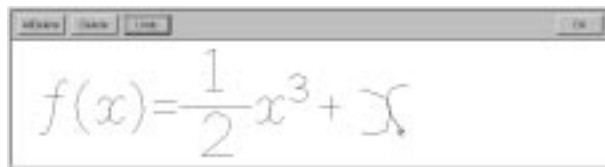
Our Handwriting Area has four menu buttons ⟨Clear⟩, ⟨Delete⟩, ⟨Undo⟩ and ⟨OK⟩. When a user pushes the ⟨OK⟩ button, the mathematical expressions written on the Handwriting Area are recognized and inserted to the Display Area of Infity at the current position of its cursor. The behaviors for the other buttons are described in a subsection below.

#### 3.1 Automatic Rewriting Method

One of the characteristic points of our handwriting input system of mathematical expressions is the automatic character rewriting method. Each written character<sup>1</sup> is rewritten by neat strokes in an appropriate size and position, immediately and automatically.

There are several reasons for adopting this method. The distortion of the input characters and the turbulence of the positions or the scales of characters usually cause serious difficulties in the structure analysis of mathematical expressions, in which the positions and the scales of the characters have special meanings. An error of a character recognition or an error of the segmentation of the strokes into character units leads sometimes to a fatal error of the structure analysis of the mathematical expression. The labors for the correction of this kind of errors disturb seriously the smooth input of mathematical expressions.

Therefore, our automatic character rewriting method is introduced in order to overcome this difficulty. In this interface, the user can identify each recognition error immediately it occurs, and can correct it easily.



The rewriting of characters proceeds as follows.

---

<sup>1</sup>Alphabets, numerals, Greek letters and other mathematical symbols are included.

The input characters are classified into two groups. One is a group of characters which can be extended to other characters by adding some strokes (=extendable character), and the other is the group of unextendable characters. For example, 'F' and 'C' are extendable since these are transformed into 'E' and 'G' by adding a stroke or strokes.

When a stroke is written, it is recognized as a character together with the untreated strokes if they exist. In the case where the recognition result is an unextendable character and the recognition cost is low (good) enough, the result of the recognition is determined and the character is rewritten neatly by computer. The strokes of the character are marked as treated strokes. Otherwise, the system waits for the input of the next stroke.

Extendable characters are also determined either when some prescribed time interval has elapsed with no input of new stroke<sup>2</sup>, or the next stroke is pushed down to a new character position<sup>3</sup>. In this way, the determination of the recognition result proceeds automatically with no supplementary operation of the user.

Since the determined character is rewritten using neat strokes into standard character form (and with the regulated size), the user can know the recognition result immediately. If the recognition result is wrong, the user can choose the correct one among three candidates of the recognition results of the character by clicking the mouse. If the correct one is not in the list of candidates, the user can rewrite them after pushing the ⟨Undo⟩ button or the ⟨Delete⟩ button described below.

Another advantage of this rewriting method is that the rewritten characters may serve as a model of the feasible character forms and the sizes for the recognition. The user learns them subconsciously. Apart from this pedagogic effect, it also leads the user to a neater writing style.

### 3.2 Menu Buttons of Handwriting Area

The Handwriting Area has four menu buttons ⟨Clear⟩, ⟨Delete⟩, ⟨Undo⟩ and ⟨OK⟩. The operations corresponding to these buttons are as follows:

1. **clear.** Clear all the characters and undetermined strokes written on the board.
2. **delete.** Delete last one character. If there is an undetermined character (untreated strokes), all the untreated strokes are cleared.

---

<sup>2</sup>The actual default value of this interval is 3 seconds.

<sup>3</sup>namely, the pushed down position contradicts to the character extension

3. **undo.** If there is an undetermined character, delete last one stroke. Otherwise, back the last character to the undetermined state.
4. **ok.** Recognize the structure of mathematical expression written on the board, and send the result to the display area. The received result is inserted into the line of the display area at the current cursor position.

### 3.3 Restrictions of Expressions

The mathematical expressions which can be recognized by our system are limited to the elementary ones such as fractions, square roots, subscripts, superscripts, integrals, limits, summations, etc., the expressions used in high school mathematics or in the first course of university mathematics. The matrices are excluded at the moment.

These structure of mathematical expressions may have the nested structures: The denominator and numerator of the fraction, for example, may have the structure of fractions, square roots, etc., the superscripts also may include fractions, and so on. However, deeply nested structure leads to the increase of small size characters and naturally increases the errors of the recognition.

As for the order of writings in mathematical expressions, almost all naturally used writing orders are allowed. For example, either the numerator or the denominator can be written first in the fractions, and either the super index or sub index may be written first in the integrals, summations etc., and so on.

The characters recognized in our system are listed below. It includes all the alphanumeric characters, some Greek letters, and other symbols frequently used in mathematical expressions.

Alphabets	a,b,...,z,A,B,..., Z
Numerals	0,1,2,...,9
Greek letters	$\alpha, \beta, \gamma, \delta, \epsilon, \theta, \pi$ , etc.
Special symbols	$\sqrt{\quad}, f, \Sigma, \infty, \partial$
Operators	$+, -, \times, \div, /$
Relational operators	$=, \equiv, \neq, \not\equiv, <, \leq, >, \geq, \rightarrow$
Function names	lim, log, sin, cos, tan
Parenthesis	(, ), {, }, [, ]

### 3.4 Distinction of Similar Characters

In the characters and symbols used in mathematical expressions, there are several pairs which can not be distinguished from each other by their own

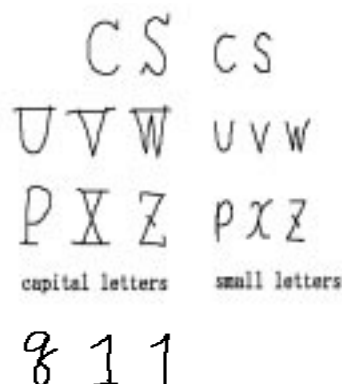


shapes. Besides the cases of capital letters and lower case letters of the alphabet such as  $C$  and  $c$ ,  $S$  and  $s$ ,  $Z$  and  $z$ , etc., there are, for example,  $q$  and  $9$ ,  $O$ (alphabet) and  $0$ (number),  $1$ (number) and  $|$ (symbol), etc.

In our system, in order to “avoid” the confusion of these symbols, the users are requested to write these special capital letters and some other characters in a designated way, as it is described below, the same way as it is used widely in Japan when a mathematical expression is written on blackboards.

A model of user interface improves considerably the performance of the system to the level of practical use, by asking for some cooperation by the user.

Character	Writing method
C, S	Start with the vertical rod
U, V, W	Attach a crossbar on the top
P	Put a crossbar at the bottom
X	Attach a crossbar both on the top and at the bottom.
Z	Start and end by a vertical rod
q(1-stroke)	Give a serif to back
1(numeral)	Start with an oblique line.



#### Stress marks for some characters

The distinction of the capital letter  $O$  and the number  $0$  is difficult without entering into the context. In our system, any circle is recognized as the numeral  $0$  if it is vertically long, and as 'o' if the aspect ratio is near to 1. When the recognition result is the numeral  $0$ , the system returns always the capital letter  $O$  as its second candidate. Therefore, the user can input the capital  $O$  by the mouse clicking on the character image of  $0$ .

The lower case letter "s" and the integral symbol has the same shape if they are normalized into a square box, but no special method is introduced in order to distinguish them, since they can be distinguished by the aspect ratio.

## 4 Conclusions

In this paper, we introduced our tentative on-line recognition system of handwritten mathematical expressions. We proposed the "automatic rewriting method" to improve the accuracy of the structure analysis of handwritten mathematical expressions and to realize an easy and prompt correction

method of recognition errors. We emphasized that an easy correction method of the recognition results is extremely important to realize smooth writing of mathematical expressions.

In order to show the possibility of its practical use, we proposed also a front tool for a computer algebra system Mathematica, having a handwriting input interface using our on-line recognition system of handwritten mathematical expressions and making use of Mathlink as the protocol. The system is implemented by Visual C++ and works on Windows 95/98. In our experimental system "Infty", we can edit mathematical expressions by hand, making use of calculation results by Mathematica as the need arises. The edited mathematical expressions can be saved into a file in the notations of LaTeX and Mathematica. Output format can be changed easily to Maple V, MathML, etc. by replacing some filtering primitive functions.

Presently, there is no convenient, inexpensive hardware appropriate to input characters by hand into a computer. Widely used data tablets to input free lines into a computer is not convenient to write characters. Hopefully, hardware to input characters by hand into computers progress further.

## 5 References

- [1] D. Blostein and A. Grbavec, "Recognition of Mathematical notation", Handbook of Character Recognition and Document Image Analysis (1997) 557-582.
- [2] T. Sakurai, Y. Zhao, H. Suguura and T. Torii, "A Front-end Tool for Mathematical Computation and Education in a Network Environment", Proc. 3rd. Asian Technology Conference in Mathematics, Springer (1998) 197-205.
- [3] Y. Sakamoto, M. Xie, R. Fukuda and M. Suzuki, "On-Line Recognition of Handwriting Mathematical Expression Via Network", Proc. 3rd. Asian Technology Conference in Mathematics, Springer (1998) 271-279.
- [4] I. Sou, M. Xie, R. Fukuda and M. Suzuki, "On Line Recognition of Handwritten Mathematical Formulae Using Stroke Relations" (in Japanese), Technical Report of IEICE. PRMU98-137 (1998) 9-16