数理解析研究所講究録 1395 巻 2004 年 252-258

Mathematical Knowledge Browser*

中川康二
KOJI NAKAGAWA
九州大学 数理学研究院
FACULTY OF MATHEMATICS,
KYUSHU UNIVERSITY†

鈴木昌和
MASAKAZU SUZUKI
九州大学 数理学研究院
FACULTY OF MATHEMATICS,
KYUSHU UNIVERSITY‡

Abstract

We propose a mathematical knowledge browser which helps people to read mathematical text. In the browser existing printed materials can be scanned and recognized by OCR (Optical Character Recognition). One of technologies needed to make the browser ideal is a method to extract automatically the logical structures and links like definitions, assertions, equations, proofs, citations from documents after OCR. In view of future development towards this goal, this paper discusses the method to extract the logical structures and links.

1 Introduction

Computers became indispensable devices for mathematics. This phenomenon can be seen by the success of mathematical systems (e.g. Mathematica or Maple) which have been being used for various other fields: physics, economics etc.

In order to apply mathematics to the real world, mathematical knowledge should be stored in computers in a way that people can easily use. Since most of the mathematical knowledge is stored in papers or books, digitizing mathematical text is becoming more and more important.

1.1 Digitized Mathematical Document

There are several kinds of digitization of mathematics. In the paper [Ada03], Adams gave some classifications of digitization of mathematics. Based on this consideration, in this paper we introduce 5 levels of mathematics digitization.

- level 1: bitmap images of printed materials (e.g. GIF, TIFF),
- level 2: searchable digitized document (e.g. PS, PDF),
- level 3: logically structured document with links (e.g. HTML(+MathML), LaTeX),
- level 4: (partially) executable document (e.g. Mathematica, Maple),

*This work is (partially) supported by Kyushu University 21st Century COE Program, Development of Dynamic Mathematics with High Functionality, of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

†nakagawa@math.kyushu-u.ac.jp
‡suzuki@math.kyushu-u.ac.jp

- level 5: formally presented document. (e.g. Mizar[miz], OMDoc[Koh00])

Currently most of mathematical knowledge is stored and used mainly in printed materials (level 1) like books or electronic journals. For being used actively it is preferable that mathematical text is stored in possibly a higher level of digitization. However since making documents digitized to a higher level needs quite a lot of efforts, the digitization of mathematical knowledge has not been enhanced so far. Therefore we definitely need software in order to automatize the digitization process in a possibly higher level.

1.2 Technologies for Automatization

The automatization can be achieved step by step between levels:

- level 1 to level 2: OCR (Optical Character Recognition),
  In order to retrieve searchable digitized document from bitmap images, OCR is used. With OCR, character sequences can be recognized from bitmap images and then they can be used for searching words. Especially recognition of mathematical formulae is the most important point. The mathematical formulae recognition has been well investigated[STF+03].

- level 2 to level 3: Extracting Structures and Links,
  Obtained data after OCR are basically characters having positions in a page structured by lines and areas. They do not directly contain meta-information (e.g. author, title) of a paper and structural information (e.g. section, subsection, itemize). Also they do not have links, which point to internal and external document. The methods to extract structures and links are the focus of this paper and will be described in Section 3.

- level 3 to level 4: Semantics Recognition from Presentation,
  Sometimes executable blocks (e.g. mathematical expressions, algorithms) appear in mathematical text. In level 3 mathematical expressions are described in the 2-dimensional (presentational) way. We need to extract semantic expressions from these presentational expressions. Mathematica[Wol03] has all standard collections of these transformation rules which retrieve semantics of presentational expressions and the user can even define their own style of notation (See `MakeExpression` function in [Wol03]).

- level 4 to level 5: Understanding Mathematical Document,
  Usually mathematical statements like definitions, lemmata, theorems, proofs are written in natural languages in books or papers. Therefore for treating them in computers we need natural language processing. The first step of the natural language processing is parsing. For parsing it is common to make a corpus which is a set of grammar rules extracted from used expressions. Making a corpus for mathematical statements was done by Baba and Suzuki[BS03]. After parsing, formalizing written mathematical description to logical formulae in a predicate logic can be achieved. Formalized statements can be used for proving in computers by theorem provers like Theorema[BDJ+01] and translated to other natural languages, e.g. Japanese. Additionally filling gaps between written informal proofs and formal proofs is an interesting topic.

Since current our mathematical activities range over all digital levels, we need a software which covers all aspects of these technologies from scanning to proving in a coherent manner. The ultimate goal is that scanned papers are processed and the software system gives us whether proofs are correct, though this goal is very ambitious.

In this paper we propose a mathematical knowledge browser which covers from level 1 to level 3. In the next section, we will discuss the mathematical knowledge browser.

2 Mathematical Knowledge Browser

The mathematical knowledge browser helps people to do mathematics from all aspects of activities in mathematics. The browser consists of three panes: structure pane, reference pane, browsing pane (See Figure 1).

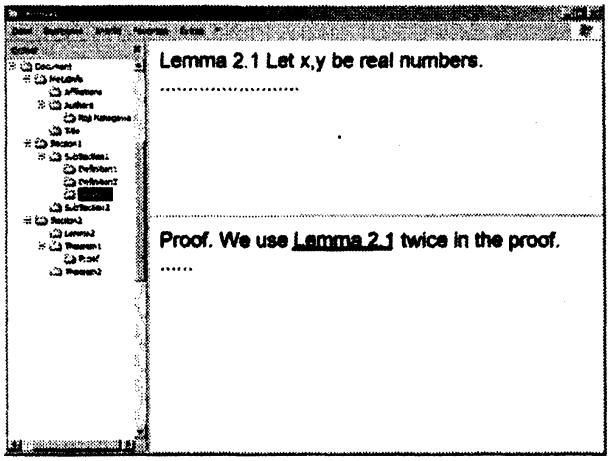

Figure 1: Screen Image of Mathematical Knowledge Browser (Sketch)

In the structure pane located in the left side, structural information (not only sections, but also structures of formulae) is shown as a tree like a file manager. The browser pane on the right bottom and the reference pane on the right top show mathematical text. By clicking a link in the browser pane the text pointed by the link will be shown in the reference pane so that people won't lose the attention in the browser pane. For example, by clicking a link 'Lemma 2.1' the reference pane shows the 'Lemma 2.1' while the browser pane does not change.

The browser can be used also for an editor for mathematical documents. With the editor one can input mathematical statements like theorems and connect the formalized formula and the statement of the corresponding mathematical statements. The formalized formulae can be sent to computing, solving, proving services located in the internet and retrieve the results. Namely it can be used for also a front-end for mathematical services. The data can be converted to other formats like LaTeX, PDF, HTML(+MathML) etc.

For building such a browser there is a good basis software called Infty Editor[1]. The editor can show the ordinary 2-dimensional mathematical notation and have the functionality of OCR.

[1] http://infty.math.kyushu-u.ac.jp/index-e.html

3 Obtaining Structured Document with Links after OCR

In this section we focus on the digitization from level 2 to level 3, namely extracting meta-information, structures, and links from mathematical documents after OCR.

3.1 Meta-Information and Structures of Mathematical Documents

The digital documents after processing OCR contain information represented by a nested structure. A document contains several pages and each page contains several areas which have positions and sizes in the page. An area can contain lines which also have their positions and sizes. A line has recognized characters with positions, sizes and font styles.

As we see, data of documents after OCR do not have meta-information (e.g. author, title) and structural information (e.g. section, subsection, itemization). The information should be automatically read off from the data.

A typical mathematical document contains the following information:

- title, authors, affiliations,
- page number,
- chapter, section, subsection,
- itemization,
- definition, lemma, proposition, theorem,
- references.

3.2 Methods for Extracting Meta-Information and Structures

In order to extract meta-information and structures, we can use positions, styles of fonts (size and style), and keywords. At first we put marks to some of lines shown in Table 1. If putting marks is done, extracting meta-information or structures is straightforward.

Here we describe observations which can be used for the methods to extract the information.

title, author, affiliations

A title usually appears on the top of the first page with centered and is presented with a bigger size bold font. The first characters of words are capitalized in a title. Authors come below the title and then their affiliations come.

page number

A page number appears as an area and consists of numbers or some Greek style like i, ii, iii, iv. The page number may appear on the bottom of a page or upper right or upper left. However it won't appear in the middle of a page.

mark	meaning	criteria of detection
TI	title	the centered largest font in a page on the top
AU	authors	names
AF	affiliations	place etc.
PN	page number	numbers
BD	beginning line of Definition	starting from 'Definition'
ED	end line of Definition	style change
BT	beginning line of Theorem	starting from 'Theorem'
ET	end line of Theorem	style change
BL	beginning line of Lemma	starting from 'Lemma'
EL	end line of Lemma	style change
BP	beginning line of Proof	starting from 'Proof'
EP	end line of Proof	ending with '□' or 'Q.E.D.'
S	section line	large fonts and starting from numbers and periods
SS	subsection line	big fonts and starting from numbers and periods

Table 1: Detection Marks for Lines

chapter, section, subsection

These entries usually start from some numbers separated by periods with a bigger and bold font. If '2 $\cdots$' is recognized as a section, '2.3 $\cdots$' should be recognized as a subsection. Also the first characters of words are capitalized like as a title.

definitions, lemma, proposition, theorem, example, exercise

These areas start with specific keywords, like Theorem, Definition, etc., starting from the beginning of lines.

A proof can start with the keyword 'Proof.' and end sometimes with '□' which expresses the termination of a proof. In the case which '□' does not appear, we may be able to detect it from the space between paragraphs.

In Table 1, the criterion 'style change' is more difficult to detect than others. For example, in the following text

LEMMA 4.1. i) *If $x \in H^*(G/H; k)$ is transgressive with respect to the bottom fibering, then the element $p^*(x) \in H^*(G; k)$ is universally transgressive.*
ii) *If $x \in H^*(G; k)$ is universally transgressive then so is $j^*(x) \in H^*(H; k)$.*
iii) *If $H^i(G/H; k)=0$ for $i<n$, $\deg x < n-1$ for $x \in H^*(G; k)$ and if $j^*(x)$ is universally transgressive, then x is also universally transgressive.*
These follow from the naturality of the transgression.
The following result is due to Borel [4] (see also Baum-Browder [2]).
LEMMA 4.2. *We can choose generators $a, x_7, x_{11}, \cdots, x_{8n+1} \in H^*(PSp(2n+1); Z_2)$ such that $H^*(PSp(2n+1); Z_2) = Z_2[a]/(a^4) \otimes \Lambda(x_7, x_{11}, \cdots, x_{8n+1})$ where $\deg a = 1$ and $\pi^*(x_{4i-1}) = e_{4i-1}$, $i=2, 3, \cdots, 2n+1$, for the projection $\pi : Sp(2n+1) \to PSp(2n+1)$.*

the first line and the fifth line should be marked as 'BL' and 'EL' respectively. Marking 'BL' is easy, because it starts from the keyword 'Lemma'. However marking 'EL' is difficult, because it should detect

the change of style. In this case, 'Lemma' is written in the italic style till the fifth line and from the sixth line it turns to the normal style. Therefore the fifth line should be marked as 'EL'. Since any papers do not write 'Lemma' in italic, it should be detected on a case-by-case basis. Another criterion can be indentation or space between lines.

Additionally formatting information of journals can be used. Since a journal usually has own uniform formatting style, we can know the position of title and page numbers etc. Journal names can be detected automatically from the first page of the paper.

3.3 Extracting Links

One of the advantages of treating mathematical document in computers is linking. Since scanned data do not have linking information, it would be nice to recognize links automatically. Possible types of linking are as follows:

chapter, section, subsection

In text sometimes chapters or sections are referred. For example, the sentence "This concept will be described in Section 2." can appear in text. Then the "Section 2" should have a link to the description of "Section 2". This can be detected by special keywords like 'chapter', 'section' with numbers.

formulae numbering

Formulae are numbered in order to be referred. Formulae numbers are located in the left or right of the formulae.

references

References of a paper are links to other papers which are related to the paper. In an electric version, it is possible to make links to the electronic papers or at least information of the papers like reviews[Mic03].

It is also possible to make a citation index automatically. There is a famous autonomous citation index called Citeseer[2)][LGB99].

mathematical technical terms

There are many mathematical technical terms like 'real number', 'group', 'ring' etc. It would be nice to link these terms to mathematical repositories, for example MathWorld[3)]. Then the readers can easily know or recall the notions without going to physical libraries.

[2)] http://citeseer.nj.nec.com/cs

[3)] http://www.mathworld.com

4 Conclusion

A mathematical knowledge browser which helps people to read mathematical text, compute mathematical expressions, and prove mathematical statements was proposed. Also the methods to extract structure and links were discussed. The techniques of extracting structures and links can be used for not only documents after OCR, but also many files in PDF. Therefore it is possible to make a PDF file indexed by sections and links from an ordinary non-indexed PDF file.

The mathematical knowledge browser can be used:

- for reading scanned papers with automatically structured and linked,
- for formalization with the help of natural language processing for mathematical text,
- as an editor which can be used for making structured mathematical documents like scientific papers, text books, and course materials,
- for a user interface front-end for mathematical engines including proving facility.

The future work is to implement the idea explained in this paper.

References

[Ada03] Andrew Adams. Digitisation, representation and formalisation: Digital libraries of mathematics. In *Mathematical Knowledge Management (MKM 2003), Feb. 16th - 18th, 2003, Bertinoro - Italy, LNCS 2594*. Springer, 2003.

[BDJ+01] B. Buchberger, C. Dupre, T. Jebelean, F. Kriftner, K. Nakagawa, D. Vasaru, and W. Windsteiger. The theorema project: A progress report. In *Symbolic Computation and Automated Reasoning*, pages 98–113. A.K. Peters, 2001.

[BS03] Yusuke Baba and Masakazu Suzuki. An annotated corpus and a grammar model of theorem description. In James Harold Davenport Andrea Asperti, Bruno Buchberger, editor, *Second International Conference, MKM 2003, Bertinoro, Italy*, pages 93–104, Feb. 2003.

[Koh00] Michael Kohlhase. OMDoc: An infrastructure for OpenMath content dictionary information. *SIGSAM Bulletin (ACM Special Interest Group on Symbolic and Algebraic Manipulation)*, 34(2):43–48, 2000.

[LGB99] Steve Lawrence, C. Lee Giles, and Kurt Bollacker. Digital libraries and Autonomous Citation Indexing. *IEEE Computer*, 32(6):67–71, 1999.

[Mic03] Gerhard O. Michler. How to build a prototype for a distributed digital mathematics archive library. *Annals of Mathematics and Artificial Intelligence*, 38:137–164, 2003.

[miz] The mizar system. http://mizar.org.

[STF+03] Masakazu Suzuki, Fumikazu Tamari, Ryoji Fukuda, Seiichi Uchida, and Toshihiro Kanahori. Infty — an integrated ocr system for mathematical documents. In *ACM Symposium on Document Engineering (DocEng '03), Grenoble, France, Nov. 20-22*, 2003.

[Wol03] Stephen Wolfram. *The Mathematica Book, Fifth Edition.* Wolfram Media, Inc., 2003.