latex2nemeth: A direct LaTeX-to-Braille transcribing tool

Andreas Papasalouros, Antonis Tsolomitis

Abstract

The Braille system allows the tactile representation of characters in various alphabets, giving access to reading texts to visually impaired persons. The Nemeth code allows the representation of mathematics symbols and expressions into the Braille system.

We have developed a tool, named latex2nemeth, for the reliable transcription of LaTeX documents to Nemeth Braille, thus facilitating the access of visually impaired students to studying science. In order to support the extensive set of mathematics symbols covered by TeX, we have proposed some new symbols based on the extension mechanisms of the Nemeth code. With the aid of latex2nemeth, we have created a repository of learning material in Braille/Nemeth code aiming to support studies in mathematics for visually impaired students. While most of the material available in the repository is in the Greek language, the tool supports other languages as well. latex2nemeth is currently available in both the TeX Live and MiKTeX distributions.

1 Introduction

In 2014 we came to know about a visually impaired student of mathematics at the University of Athens. The student could read and write in Braille notation and in the Nemeth representation of mathematics. Although the lecture notes and course textbooks were available in LaTeX source format, there was no reliable way to translate these sources into tactile representations, accessible by visually impaired persons. The student had to manually transcribe learning content with the help of a seeing person. We set out to create a tool for transcribing LaTeX texts to tactile representations accessible to blind persons. The above is an instance of a wider problem: Visually impaired students of related fields did not have access to the bulk of study material available in LaTeX format.

2 The Braille and Nemeth codes

2.1 The Braille code

The Braille system allows the tactile representation of characters in various alphabets, giving access to reading texts to visually impaired persons. The six-dot Braille system supports the representation of $2^6 = 64$ different characters. An assignment of Braille symbols to letters defines a certain Braille alphabet (e.g., English, Greek, etc.).

Figure 1: Reading of Braille/Nemeth code

2.2 The Nemeth code

The Nemeth code allows the representation of mathematics symbols and expressions in the Braille system. It defines a set of *rules* that designate the combinations of Braille symbols that describe various types of mathematical structures.

Nemeth's original document [4] specifies 25 rules covering various aspects of mathematical structures such as: numeric signs and symbols, alphabets, fractions, superscripts and subscripts, radicals, symbols of grouping, spatial arrangements, etc.

3 The latex2nemeth tool

The aim of the latex2nemeth program is twofold:

- The *reliable* transcription of books and electronic notes from LaTeX to Nemeth/Braille.
- The creation of a repository of mathematical texts available to visually impaired students and researchers.

At least at the beginning of this project (end of 2014), to the best of our knowledge, no tool or method met the above requirements.

3.1 Existing solutions

At the time of the design of this project, few solutions existed for transcribing LaTeX documents into Braille.

An available commercial solution implied the following steps: 1) conversion from LaTeX to MS Word with MathType; 2) conversion from MS Word to Braille with the use of the Duxbury software.

The above method supports only a small subset of LaTeX commands, thus, the process produced unreadable Braille code with many mistakes (through extensive tests during 2014).

An open source solution with the use of the liblouis library [3] was also extensively tested in the initial phases of this project. This solution had the following steps: 1) conversion from TeX into MathML (e.g., with TeX4ht [6]); 2) conversion

doi.org/10.47397/tb/42-2/tb131papasalouros-latex2nemeth

from MathML into Braille with the liblouis library. This process also created Braille code with many errors, given that the liblouis library did not aim at supporting TEX at the time of the creation of the latex2nemeth program.

3.2 Program features

The main features of the latex2nemeth software are the following:

- LATEX files with text in Greek, English and Ancient Greek are converted to Braille.
- More than 850 different mathematical symbols and expressions are supported.
- All TEX AMS mathematical symbols are covered, among others.
- Different Braille alphabets can be supported with the use of different *symbol tables*.
    - A symbol table is a JSON file that maps individual characters or TEX commands to Braille characters or sequences of characters.

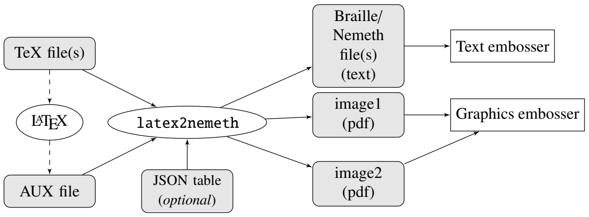


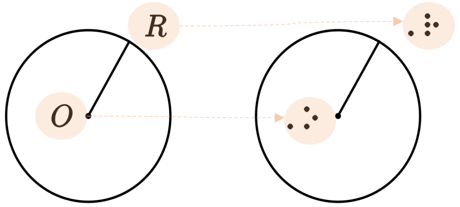
Figure 2: Flow of translation to Braille/Nemeth

The process of generation of Braille text is illustrated in figure 2. The latex2nemeth tool takes as input a source TEX file, or set of files, as well as the corresponding .aux file. The latter is used for expressing in Braille numbered elements such as figures, sections/subsections, mathematical equations, etc. The output of the tool is a set of Unicode (UTF-16) text files with Braille characters. These files are first converted to UTF-8 by an appropriate tool (such as iconv) and then imported to LibreOffice using a font that supports 6-dot Braille (such as NewComputerModern) and printed in tactile six-dot Braille form in a specialized text embosser with the use of odt2braille LibreOffice extension. The above process is automated by an appropriate script, available in the tool distribution.

In its default configuration, the tool transcribes texts that are written in English or in modern Greek. As seen in figure 2, the tool takes as an optional input a JSON table file. This file can be provided by the user in order to support a Braille alphabet different than the above such as Hebrew, Cyrillic, etc. As a demonstration of this extensibility feature, we provide in the standard distribution of the tool a JSON file named polytonic.json with the definition of the Ancient Greek Braille alphabet.

3.2.1 Image label filtering

The latex2nemeth tool provides limited support for converting text inside images into Nemeth format. More specifically, images defined with the PSTricks library may contain labels with mathematics expressions delimited with the symbol $. These expressions are filtered by the tool and they are translated in Nemeth, as illustrated in figure 3. The generated images are exported separately in PDF and can be printed separately in a specialized graphics embosser, as shown in figure 2.



Original code	Transcribed code
`\begin{pspicture}(-1,-1)(1.2,1.2)`	`\begin{pspicture}(-1,-1)(1.2,1.2)`
`\pscircle(0,0){1}`	` \pscircle(0,0){1}`
`\qdisk(0,0){1pt}`	` \qdisk(0,0){1pt}`
`\uput[180](0,0){$O$}`	` \uput[180](0,0){ ⠠⠕}`
`\psline(0,0)(.5,.866)`	` \psline(0,0)(.5,.866)`
`\uput[30](.5,.866){$R$}`	` \uput[30](.5,.866){ ⠠⠗}`
`\end{pspicture}`	`\end{pspicture}`

Figure 3: Transcribing image labels

4 Implementation

The transcriber is based on a parser for the LATEX language, created specifically for this project. The parser recognizes the most common LATEX commands and environments in text and mathematics modes and also covers most structures and mathematical symbols. The program is developed in Java. The lexical analyzer and the parser were developed using the JavaCC compiler generation tool [2]. The tool is available as a JAR (Java archive) package.

4.1 LATEX to Braille transcription

Each paragraph and each environment in the input LATEX sources is processed separately. In text mode, each lexical token is recognized and transcribed into its corresponding Braille symbol by using a certain *symbol table* given in the form of a JSON file, as mentioned earlier. Numbers are lexically scanned as atomic entities, e.g., 13.455, since, according to Braille code, a certain numeric indicator (⠼) must precede the whole number instead of each numerical symbol in the number.

4.2 Processing of mathematical expressions

Mathematical expressions are parsed into appropriate syntactical trees in memory. The abstract syntax trees for mathematics expressions are independent of the target language (Nemeth). Depending on the type of expression (fraction, superscript/subscript, etc.) an appropriate procedure (semantic routine) generates corresponding Nemeth code.

An object-oriented representation of mathematical expressions was adopted, based on the Composite and Interpreter design patterns [1].

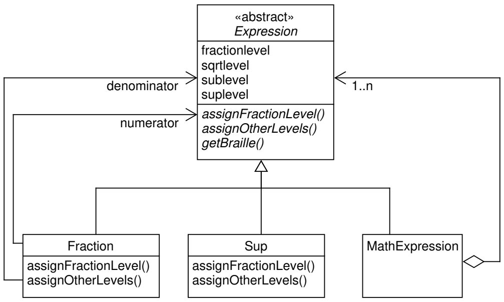


Figure 4: Class diagram of basic mathematical expression classes

4.3 Expression indicators and depth

In order to assist a blind person in conceiving the structure of a mathematical expression, in Nemeth code, an *expression indicator* signifies not only the kind but also the *depth* of certain expressions that can be nested such as fractions, subscripts and superscripts, root expressions, etc.

$$e^{x^{b+1}} + \frac{c}{d+\frac{k}{x+3}}$$

The above expression is rendered in Nemeth Braille code as shown in figure 5. In the example, the fraction and the exponent are defined by appropriate *expression indicators*. Thus,

```
e<exp>x<exp-2>b+1<base>
+
<openfrac-2>
c
<fractionbar>
d
+
<openfrac>k<fractionbar>
x+3
<closefrac>
<closefrac>
```

Figure 5: Expression indicators in nested expressions

- `<exp>` (⠘) signifies a simple superscript
- `<exp-2>` (⠘⠘) signifies a superscript within a superscript.
- `<openfrac>` (⠹) signifies the opening of a single fraction.
- `<openfrac-2>` (⠠⠹) signifies the opening of a complex fraction, etc.

Depth indicators for various expressions is calculated by appropriate methods of the instances of the *Expression* Java class, as illustrated in figure 4: the depth for complex fractions is calculated *bottom-up* by the `assignFractionLevel()` method, while the depth for other nested expressions (superscripts, subscripts, roots, etc.) is calculated *top-down* by the `assignOtherLevels()` Java method.

4.4 Spatial arrangements

Certain mathematical structures such as arrays, case and align expressions, etc., are specially aligned, in the sense that their elements are positioned in a certain horizontal or vertical arrangement. Since the Nemeth code supports spatially aligned structures, the `latex2nemeth` tool generates spatially aligned Nemeth code. The following example defines an `array` structure:

```
$$\chi_F (x)=\left\{\begin{array}{ll}
  1, & \textrm{if } x\in F\\
  0, & \textrm{if } x\notin F
  \end{array}\right.$$
```

This structure corresponds to the definition:

$$\chi_F(x) = \left\{ \begin{array}{ll} 1, & \text{if } x \in F \\ 0, & \text{if } x \notin F \end{array} \right.$$

For the above, the tool generates the following spatially aligned Braille code:

```
  ⠨⠯⠰⠠⠋⠐⠷⠭⠾⠨⠅
⠠⠨⠷⠼⠂⠠  ⠊⠋⠭⠈⠑⠠⠋ ⠠
⠠⠨⠷⠼⠴⠠  ⠊⠋⠭⠌⠈⠑⠠⠋ ⠠
```

5 Extension of the Nemeth code: Proposing new symbols

The Nemeth code prescribes certain mechanisms for defining new symbols. As an example, the symbol $\hookrightarrow$ (`\hookrightarrow`) is defined based on the plain arrow, expressed as ⠒⠒⠕ in Nemeth. The new, modified, Braille symbol is ⠫⠈⠿⠒⠒⠕ This new symbol:

- Starts with character ⠫ that signifies grouping of characters in a new symbol.
- Contains the combination ⠈⠿ that signifies the characteristic hook.

To support the extensive set of symbols used in university-level mathematical texts, by using extension mechanisms such as the above, we have proposed other new Braille mathematical symbols as well.

6 Evaluation — Reliability of transcription

A formal evaluation has been conducted [5] using mathematical documents from the AMS. The quality of the transcription was evaluated by comparing the transcribed documents with back translations generated by two visually impaired persons with good knowledge of the Nemeth code. The average error of the two back translations was 2.4% while the inter-rater agreement was very high (Cohen's kappa ≈ 0.98).

The tool fails to discriminate among symbols *period* and *decimal point* in the TEX sources, which results in a small number of errors, given that these symbols have different representations in Braille/ Nemeth, but the same representation in TEX (`.`). Also, the tool adds a numeric indicator before every number, while, according to the Nemeth rules, some numeric indicators should be omitted in certain numeric expressions. Although this output does not strictly conform to the Nemeth rules [4], the redundancy of some generated numeric indicators was found not to confuse the reader, and result in only a slight increase in the Braille text size.

As an informal evaluation, much of the generated content has been used as study material by the blind student mentioned before. The student has graduated on schedule and now she is pursuing graduate studies with the help of `latex2nemeth`.

7 Distribution and documentation

The website of the `latex2nemeth` project is: `myria.math.aegean.gr/braille/index-en.html`. The project can also be accessed via the CTAN repository at `ctan.org/pkg/latex2nemeth`, and its source code can be accessed at `sourceforge.net/projects/latex2nemeth`. The `latex2nemeth` program is currently available in the TEX Live, MiKTEX and MacTEX distributions.

The following documentation is available at the project website:

- Dictionary of mathematical terms (from symbol name to Nemeth).
- Reverse dictionary of mathematical terms (from Nemeth to symbol name).

The above dictionaries are available in both printed and Braille forms, so they are accessible by both seeing and visually impaired persons.

7.1 Towards a mathematical/scientific repository for the visually impaired

With the use of the tool we are creating a repository of publicly available mathematical/scientific content in Braille code. So far, the following complete texts have been transcribed: 1 book in English, 23 books and course notes in Greek, and 2 books in ancient Greek: Homer's Odyssey and The Orphic Hymns. All books are available via the project's website.

8 Limitations and future work

The tool has limited support for macros. Currently a simple parameter substitution mechanism is implemented. Thus, we plan a full re-implementation with full macro expansion support.

The discrimination among symbols *period* and *decimal point* is planned to be resolved in a forthcoming version. Also, we are working on avoiding redundant numeric indicators, as described above, by applying the appropriate Nemeth rules based on deep parsing of numerical expressions.

We are also working on more extensive support for images, beyond the simple filtering of image labels for PSTricks figures mentioned above.

Acknowledgments

We would like to thank Mrs. Olga Maleza for the support and the information that she has provided from the beginning of this project.

The development of the tool was partially supported by the University of the Aegean Research Unit (grant no. 2625).

References

[1] E. Gamma, R. Helm, et al. *Design Patterns: Elements of Reusable Object-oriented Software.* Addison-Wesley, Boston, MA, USA, 1995.

[2] The JavaCC compiler compiler. `javacc.github.io/javacc`.

[3] The `liblouis` project. `liblouis.org`.

[4] A. Nemeth. The Nemeth Code 1972. `brailleauthority.org/mathscience/nemeth1972.pdf`.

[5] A. Papasalouros, A. Tsolomitis. A direct TEX-to-Braille transcribing method. *Journal of Science Education for Students with Disabilities* 20(1):36–49, 2017. RIT Scholar works.

[6] The `tex4ht` project. `tug.org/tex4ht`.

◇ Andreas Papasalouros,
Antonis Tsolomitis
Department of Mathematics,
University of the Aegean
832 00 Karlovassi, Greece
`andpapas (at) aegean dot gr`,
`atsol (at) aegean dot gr`