CrossMark

Multimedia platform for mathematics' interactive learning accessible to blind people

Michał Maćkowski[1] · Piotr Brzoza[1] · Marek Żabka[2] ·
Dominik Spinczyk[3]

Abstract Nowadays, the math learning is an important step in developing professional carriers in technical and economic sciences. Increasing the number of e-learning tools used in universities courses can reduce the potential barrier of access to mathematical knowledge, but most of them are not accessible for impaired students. Moreover, classical printed math books include little explicit instructional information about structural information interpretations. Taking into account these barriers the article presents the developed method used for creating interactive steps of decomposed math's exercise solution and alternative description of math formulas accessible for the blind. The elements of proposed methodology: generation of state machine, design and presentation of transition conditions, generating the presentation layer and a typical usage by a blind user are presented. A set of rules for describing mathematical formulas were proposed after consultation with mathematicians and teachers of blind people. The application was developed as web application. The graphical interface of presented application was designed using PHP and JavaScript technologies. The collection of prepared exercises include about 240 prepared exercises from different areas of mathematics

✉ Dominik Spinczyk
dominik.spinczyk@polsl.pl

Michał Maćkowski
michal.mackowski@polsl.pl

Piotr Brzoza
piotr.brzoza@polsl.pl

Marek Żabka
marek.zabka@polsl.pl

[1] Faculty of Automatic Control, Electronic and Computer Science, Silesian University of Technology, 16 Akademicka, 44-100 Gliwice, Poland

[2] Faculty of Applied Mathematics, Silesian University of Technology, 23 Kaszubska, 44-100 Gliwice, Poland

[3] Faculty of Biomedical Engineering, Silesian University of Technology, 40 Roosevelta, 41-800 Zabrze, Poland

⌖ Springer

and 60 selected exercises including alternative description layer. About 1000 students and about 40 impaired students, from 6 faculties of the university use this platform during math courses for both self and class learning. The defined rules were used to read aloud mathematical formulas to the visually impaired people with a different level of mathematical knowledge. The results confirmed good understanding of mathematical formulas by using prepared alternative description.

Keywords Interactive multimedia math presentation · Alternative math presentation · Math tutoring platform · Distance learning

1 Introduction

Presentation of the structural information contained in math formulas is a major challenge in the field of interactive distance math training for visually impaired people. Blind people use the voice presentation, but typical linear presentation with the voice may be ambiguous. This problem has been partially solved in the mathematics extension of Digital Accessible Information System (DAISY) standard (ANSI/NISO Z39.86-2005 R2012 Specifications for the Digital Talking Book) [20]. The mathematical formula is written in MathML notation with additional XML tag, which contains a link to the alternative voice description in the audio file. The standard does not define the principles of an audio alternative description for math formulas and does not allow navigation the inside description [4, 25, 32, 36]. Currently existing solution is EasyReader that allows to read aloud mathematical formulas in DAISY book, without navigation capabilities [10]. Another solution is MathPlayer that supports IE, Firefox, Word and PowerPoint. It can read aloud mathematical *formulas* written in MathML notation using speech synthesizer. Moreover, the supported screen readers must be installed in operating system (JAWS or NonVisual Desktop). This tools provides 3 navigation modes [6, 27, 28]:

- enhanced - navigation by mathematically meaningful pieces (operators, delimiters, and operands),
- simple - navigation by words except when you get to a 2D notation (fractions, roots, etc.), then it speaks the entire notation,
- character - navigation by words or characters.

One of the most important stages in mathematical education is self-solving math exercises. Disabled people encounter additional difficulties in accessing to visually presented information and communication with others.

There are solutions for computer-aided presentation and learning of mathematics. One of the most popular presentation tool is MathJax. This tool is a JavaScript library for visual rendering of formulas created in MathML, LaTeX and ASCIIMathML using any operating systems and web browsers [5]. One example of education platform is ALEKS [7]. The idea of using these tools is that first the student's knowledge is evaluate by acquired answers to the questions. Based on the assessed responses ALEKS chooses the correct subject and the degree of difficulty of presented materials. Another tool supporting math learning is multi language e-learning platform ActiveMath [17]. It integrates several features and tools e.g. computational algebra, functions chart plotter,

search and user annotation. Other example of popular e-learning platform is Khan Academy [22], which offers courses from many disciplines, including mathematics. However, the visually impaired users report the accessibility problems (e.g. the problem with reading numbers by screen readers like JAWS or NVDA). Accessibility of these platforms was evaluated by visually impaired people using assistive technologies. The results of the research conducted by the authors show low level of accessibility.

There are also other tools dedicated to impaired pupils. One of the existing example is MiniMatecaVox [18], which provides subsidies for math teaching for the literacy phase of visually impairment children. The software presents the children a ludic and pleasant environment providing them at the same time math teaching, digital inclusion and recreation in their education initial phase. More general solution is ChromeVox, which is accessibility plugin to Chrome web browser [31]. It supports in particular the smart verbal presentation and exploration of content that is either given in MathML or rendered with Mithra. To present mathematical formulas in alternative form the Speech Rule Engine (SRE) is used. Speech Rule Engine is a JavaScript library that translates XML formulas into speech strings according to the rules that can be specified in an extended XPath (XML Path Language) syntax [30]. Currently, Sorge in his solution proposed integration of Emacspeak (enable visually impaired users to employ Emacs editor), MathJax and Speech rule engine to support visual impaired learners in editing mathematics formulas [29].

In summary, there are many challenges in presenting the structural information appearing in math exercises, in a concise way and consistent with the context. From the technical point of view, the presented solution takes into account popular assistive technologies like screen readers. The authors of Lambda project in papers [1, 23] presented the alternative and graphic presentation of mathematical formulas using Braille and computer displays. The desired solution should not contain additional requirements such as Braille monitors, because of their high cost. Additional barrier is the difference between 8-point mathematical Braille notations, proposed in this project, comparing to 6-point national mathematical Braille notations. Garderen et al. reports very little explicit instructional information about structural information representations in the classical math text books [15]. Moreover, the existing solutions do not allow interactive editing and following all the steps required to solve the exercise by students themselves. The presented solution in this paper can overcome the described challenges. By dividing the exercise into sequence of elementary step the impaired user can easily switch between details and wider context. The rules how to present the details at the specific step of the exercises were consulted with mathematicians and visually impaired students. ForMath multimedia platform, which was developed, is primarily a collection of interactive exercises tailored for the math program implemented in most faculties of technical universities [2]. The system offers two possibilities. First option refers to solving the exercise independently, and then checking the results. Comments to the wrong answers are presented in case of typical errors. Second option is to solve the exercise by using created platform. In this case, a user is getting: hints, mistakes' descriptions, theory and various paths of solution. Continuous analysis of faults in each step of the exercise being solved allows on one hand to correct solution, and on the other hand allows the system to suggest another exercise according to the needs of a particular user.

The aim of this paper is to present the developed multimedia solution. The article is organized as follows. In the section 2 the following issues are described: generation of state machine, design of transition conditions, generating the presentation and accessible description layers and a typical

usage by a blind user. The section 3 contains the examples of the generated screens corresponding to solve exercise, which is presented in previous section. The section 4 summarize the advantages of the proposed method comparing to Braille'a notation and DAISY standard.

2 Materials and methods

2.1 Representation of the information contained in math formulas

Mathematical symbols are well known de facto international standard and are included in Unicode character set [33]. Commonly used notations to describe the math formulas are Mathematical Markup Language (MathML) [16] and LaTeX [21]. MathML is a mathematical markup language, an application of Extensible Markup Language (XML) for describing mathematical notations. It is a part of HTML5 and an ISO standard ISO/IEC DIS 40314. MathML deals not only with the presentation, but also with the meaning of formula components. Some efforts are focused on the development of translation tool to process MathML into native-language text accessible for visually impaired people [26]. Some publication report that it is a difficult task [13].

The origin aim of the TeX was to create something that would allow simple construction of mathematical formulas. The strength of this notation is the math formulas presentation both in printed documents and web pages comparing to MathML notation [34].

In our application we decided to use TeX notation due to its better presentation of complex math formulas. However, some researchers report that TeX does not provide an encoding of the semantics for the mathematical material and this information must be derived from some (external) context [34]. Due to this fact, to guarantee the high quality of education materials, we decided to avoid automatic conversion to alternative text description.

2.2 Decomposition of math exercise into steps

To increase the value of the prepared materials for self- study, self-assessment and evaluation, the first step of the planned methodology is to divide the exercise into following sequence of elementary steps leading to solve a mathematical problem. The extracted steps and the conditions of the transition between them create the decision tree of the exercise solution (Fig. 1). At the every step of solving the exercise the possible types of errors are defined.

2.3 Methods of preparing visual presentation exercise

To generate elements of solved exercise the specific language was created, which consists of 5 groups of commands:

- screen and tiles images commands – responsible for generating full screens (e.g. \fmScreen, \fmWindow) and their elements (e.g. \fmTileImage),
- control's commands – responsible for generating instances of various kinds of control buttons (e.g. \fmButton, \fmNext),
- interactive element commands - responsible for generating interactive element (e.g. \fmText, \fmList, \fmOption),
- action commands – responsible for generating interactive elements (e.g. \fmGoTo),
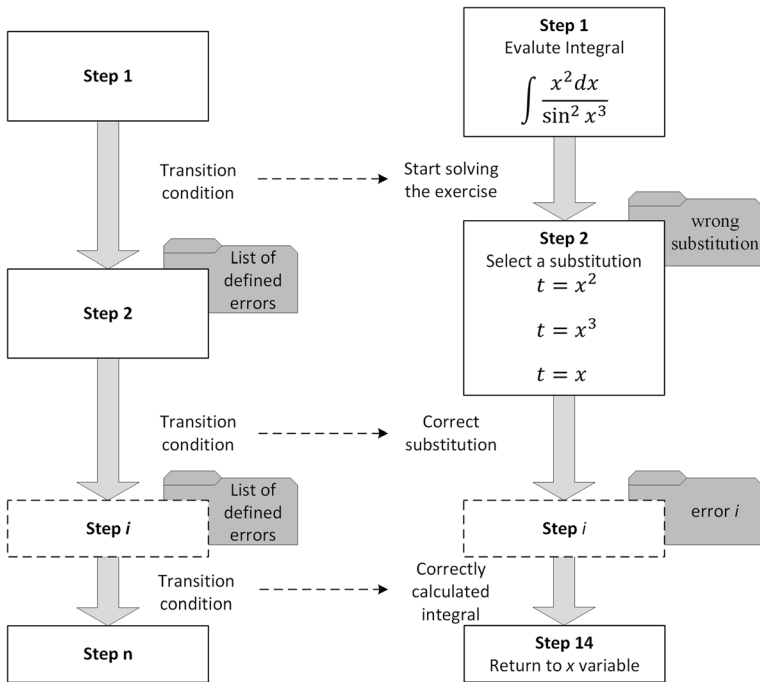
Fig. 1 The concept of the exercise decision tree (*left*) and its example (*right*)

- others commands – every other commands not included in above groups (e.g. \fmExercise, \fmEnd).
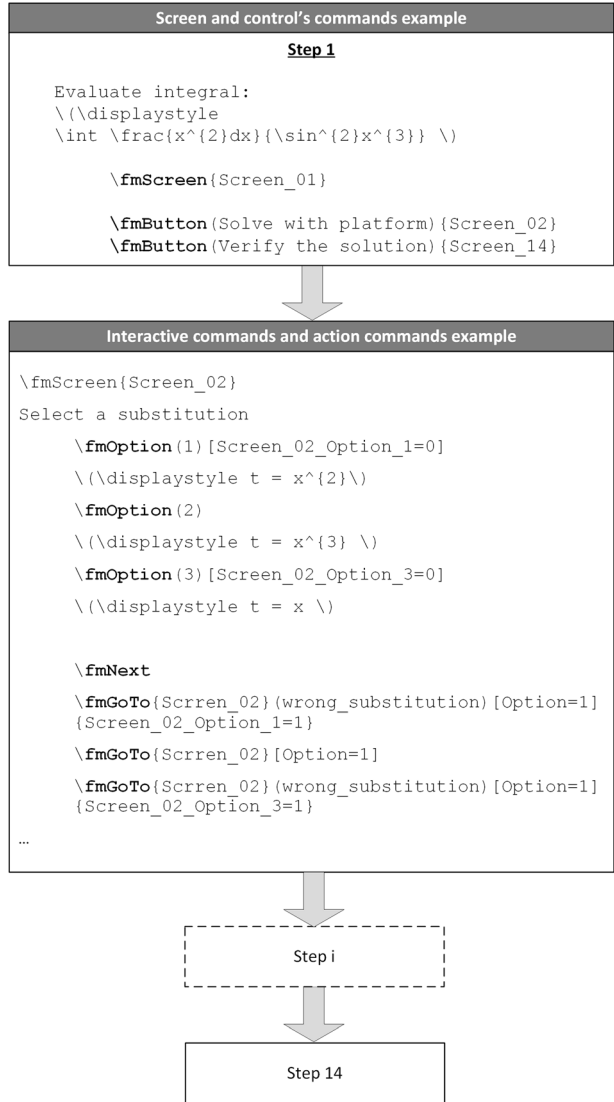
Mathematical formulas found in various stages of exercise are written in TeX notation. The example of mentioned additional commands are presented in Fig. 2.

A mathematician is preparing the graph of solving process of an exercise, with all incorrect ways. He/she must use the experience to predict student errors and for all these errors a screen with explanations and comments. Sometimes it is necessary to present some screens of theory. To establish the connection between screens the author of an exercise can use the commands \fmGoTo. It was convenient to consider a screen as a collection of tiles with a condition of visibility. Commonly each error is associated with one tile on the bottom of screen. So many of fm-commands have a condition of visibility.

The prepared TeX source is processed by program fm, written in C++ especially for this project. It divide the text into screens using \fmScreen, \fmWindow. Then the screens are divided into tiles by \fmTileImage and the button commands. Each tile is wrapped separately into LaTeX file, then the programs: latex and dvipng are executed on this file. Formally, the buttons are a kind of tiles but there is no need to proceed them by latex and dvipng. On the tile there are some interactive elements, occupies rectangular areas. The program calculates the position of them.

Then the program fm must read all variables, conditions and assignments. A special fm-language was developed to support the logic of an exercise. There are 5 types: strings, integers, Boolean values, names of screens and windows. There are also eight

Fig. 2 The decomposition of the exercise into steps: the extracted step example in TeX notation

```
          Screen and control's commands example

                        Step 1

 Evaluate integral:
 \(\displaystyle
 \int \frac{x^{2}dx}{\sin^{2}x^{3}} \)

        \fmScreen{Screen_01}

        \fmButton(Solve with platform){Screen_02}
        \fmButton(Verify the solution){Screen_14}
```

```
        Interactive commands and action commands example

 \fmScreen{Screen_02}

 Select a substitution

        \fmOption(1)[Screen_02_Option_1=0]

        \(\displaystyle t = x^{2}\)

        \fmOption(2)

        \(\displaystyle t = x^{3} \)

        \fmOption(3)[Screen_02_Option_3=0]

        \(\displaystyle t = x \)


        \fmNext

        \fmGoTo{Scrren_02}(wrong_substitution)[Option=1]
        {Screen_02_Option_1=1}

        \fmGoTo{Scrren_02}[Option=1]

        \fmGoTo{Scrren_02}(wrong_substitution)[Option=1]
        {Screen_02_Option_3=1}

 …
```

```
                        Step i
```

```
                        Step 14
```

types of variables, local and global. We construct expressions with the following actions: +, −, *,/, %, |x|. We can use comparisons: <, >, <=. >=, =, !=. The results are Boolean values for which we can use logical operations: and, or and not. Eventually, we get conditions used for describe visibility. Some fm commands have assignments: \fmOnEnter, \fmOnExit and \fmGoTo. The assignment has the form "variable = expresion". The position in source is used to distinguish comparisons from assignments. The whole fm-language is described in [37].
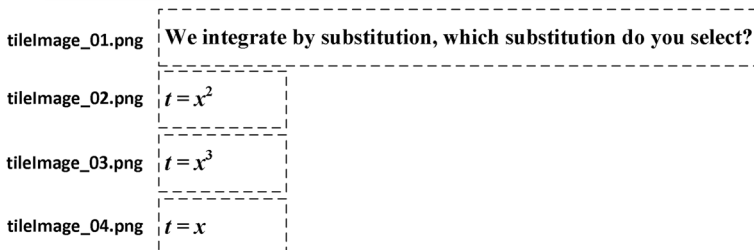
The program fm compiles the variables, conditions and assignments into JavaScript language. The example: the condition can be written as follows: $1 < x + y < 4$. We get after compilation in JavaScript: $((1 < z.x + z.y)\&\&(z.x + z.y < 4))$.

Based on the exercise description prepared in TeX notation the collection of screens representing the various stages of solution, transition conditions, types of errors and interactive elements have been generated by developed interpreter.

As a result of the interpretation of the source the following components are produced:

- the collection of tiles images for each screen – representing the rectangle areas of each screen (.eps,.png format) (Fig. 3),
- exercise structure file – including: the screens order, transition conditions, java script code to process the interactive event in CDATA markup (.xml format) (Fig. 3),
- the collection of alternative text descriptions corresponding to components of math formulas (.xml format) (Fig. 4),
- the structure includes all variables defining the exercise status (Java Script Object Notation format) (Fig. 4).

```xml
<?xml version="1.0" encoding="UTF-8" standalone="true"?>

<Exercise version="1475224393"
application_version="fm312">

<contetnt id="contetnt" rys="Exercise_Contetnt.png"
Width="98" Height="946"/>

<screen id="Screen_01" type="Screen">

<tileImage type="Button" String="Solve with platform">

<tileImage type="Button" String="Verify the solution">

</screen>

<screen id="Screen_02" type="Screen">

+<tileImage Picture="tileImage_01.png" Height="87"
Width="1103" type="normal">

+<tileImage Picture="tileImage_02.png" Height="87"
Width="946"  type="selection" value="1">

+<tileImage Picture="tileImage_03.png" Height="87"
Width="946"  type="selection" value="2">

+<tileImage Picture="tileImage_04.png" Height="87"
Width="946"  type="selection" value="3">

...

</screen>

...

</Exercise>
```

tileImage_01.png	We integrate by substitution, which substitution do you select?
tileImage_02.png	$t = x^2$
tileImage_03.png	$t = x^3$
tileImage_04.png	$t = x$

Fig. 3 Fragment of the structure file of the exercise (*up*) and generated tile images (*down*)

Fig. 4 The alternative description file (*up*) and JSON structure include variable defining the exercise status (*down*)

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>

<content>
<![CDATA[Integral of fraction nominator x power 2 end of power
denominator sinus power 2 end of power x power 3 end of power
end of fraction dx = - fraction nominator 1 denominator 2 end of
fraction cotangent x power 3 end of power + C, C is an element
of the set R]]></content>

<screen id="Screen_01" typ="screen" src="l:9:1">
<field type="button" number="kaff1-1">
<![CDATA[Solve with platform]]></field>
<field type="button" number="kaff2-1">
<![CDATA[Verify the solution]]></field>
</screen>

<screen id="Screen_02" typ="screen" src="l:70:1">
<field type="text" number="tileImage_01-1">
<![CDATA[Select the substitution]]></field>

<field type="button" number="tileImage_02-0x">
<![CDATA[Chosen x power 2 end of power]]></field>
<field type="button" number="tileImage_02-0o">
<![CDATA[Not chosen x power 2 end of power]]></field>
<field type="text" number="tileImage_02-1">
<![CDATA[           ]]></field>

<field type="button" number="tileImage_03-0x">
<![CDATA[Chosen x power 3 end of power]]></field>
<field type="button" number="tileImage_03-0o">
<![CDATA[Not chosen x power 3 end of power]]></field>
<field type="text" number="tileImage_03-1">
<![CDATA[           ]]></field>

<field type="button" number="tileImage_04-0x">
<![CDATA[Chosen x]]></field>
<field type="button" number="tileImage_04-0o">
<![CDATA[Not chosen x]]></field>
<field type="text" number="tileImage_04-1">
<![CDATA[           ]]></field>

<field type="button" number="tileImage_06">
<![CDATA[Next]]></field>
</screen>
```

```json
{
...
"e_$wrong_substitution":0,
"e_$wrong_substitution$$type":"Error",
"Screen_01$Option":0,
"Screen_01$Option_":0,
"Screen_02$Option":0,
"Screen_02$Option_":0,
"Screen_02_Option_1":0,
"Screen_02_Option_1$$type":"Global variable",
"Screen_02_Option_3":0,
"Screen_02_Option_3$$type":"Global variable",
...
}
```

For better understanding the way how the program is working, let us trace a fragment of a source:

```
\fmScreen{E2}
(…)
\fmOption(1)[E2wyb1=0]
\noindent
(…)
\fmTileImage[war=1]
```

```
This is not the correct(…)correct answer. (*)
\ fmTileImage[war=3]
```

There can be only one series of options by a screen. We use variable "war" if we refer to the selected option. The full name of the variable is E2$war, where E2 is the name of a screen and "$" is a separator. In JSON file we got:

```
(…)
"E2$war":0,
(…)
```

The tile (*) is compiled to an image "tile_image_06.png". In XML we got something like that:

```
<tileImage Width="1103" Height="127"
Picture="tile_image_06.png"
type="normal">
<conditon><![CDATA[(z.E2$war==1)]]></condition>
</tileImage>
```

There are several kinds of tile images: normal, selection, button and special button. Another collection of fields are e. g.: field of list element selection, the number input field, the string input field etc. The state machine manages to display dynamically the tiles images depending on the user interaction on every stage of solved exercise.

2.4 Interactive multimedia presentation of solved exercise in web browser

Exercise content loaded into web browser is divided into logical layers:

- active layer – contains elements of user interaction,
- graphical layer – includes graphical representation of each step,
- alternative description layer – contains alternative text description of math formulas components.

In order to enable a blind person to use prepared mathematical exercises the installation of screen reader in the operating system is required. User interaction is possible by keyboard shortcuts and screen reader commands, which allows text and mathematical formulas to be interactively read aloud.

The program formath.js. which prepare the view, work with AJAX and reads XML file. For each tile, the window in a current screen program checks the visibility condition to find out if join the tile to the screen. The values of the variables are taken form JSON file.

To make the program personal for each student, the JSON file for each exercise chosen by him/her is saved in his/her own directory. The JOSN file saves also a name of current screen and values of each variable, especially the number of errors.

Let us take closer look to the process of changing the screen.

The author simply puts down:

```
\fmButton(Next)
\fmGoTo{E2}(BB)[war=1]{E2wyb1=1}
\fmGoTo{E4}[war=2]
\fmGoTo{E2}(BB)[war=3]{E2wyb3=1}
```

In XML we got:

```
<tileImage type="button" text="Next">(…)
```

```
<goto target="E2">
<condition><![CDATA[(z.E2$war==1)]]></condition>
<assignment><![CDATA[z.Enext='E2']]></assignment>
< assignment><![CDATA[z.e_$BB++]]></assignment>
<assignment><![CDATA[z.E2wyb1=(1)]]></assignment>
</goto>
<goto cel="E4" src="l:37:33">
<condition><![CDATA[(z.E2$war==2)]]></condition>
<assignment><![CDATA[z.Enext='E4']]></assignment>
</goto>
(… one more go to …)
```

Program formath.js make "eval()" on each condition of "goto" series. If the condition is true, the program makes the assignments. Let us notice that the target screen (E2 or E4) is changed to assignment "z.Enext='E2'".

The number of errors named "BB" is saved in JSON file as "e_$BB". The PHP program chooses the next exercise for the student by examining the number of errors in local JSON files.

2.5 Alternative math formulas presentation accessible for blind user

There are no defined standards for reading aloud math formulas for Polish language. The same formulas can be read in different ways depending on math knowledge which can lead to ambiguity in the understanding of them. Voice reading are also difference according to various culture, languages and user experience.

Text descriptions used in prepared exercises were consulted with mathematicians and teacher of blind people. At the beginning of our research, the Speech Rule Engine was not in stable state. Moreover, it supports only English language [30]. To solve the problem of presenting the mathematical formulas in native language (Polish), a set of rules for describing the mathematical formulas were proposed in our own. The alternative description of new mathematical exercise, in the approach based on proposed rules, must be prepared manually. The proposed rules are collected in guideline (prepared by the authors), including the rules and usage example for different areas of mathematics. Comparison between presented approach and automatic alternative translation built in Speech Rule Engine will be presented in section 4.

Next, the defined rules were used to read aloud mathematical formulas to the visually impaired people with a various level of mathematical knowledge. The results confirmed good understanding of mathematical formulas by using prepared alternative description [3]. In the presented example of the exercise, alternative descriptions of proposed solutions are as follows:

Proposal 1

$$\int \frac{x^2}{sin^2 x^3} dx = -\frac{1}{2} \cot x^3 + C \qquad C \in R \qquad (1)$$

Integral of fraction nominator x power 2 end of power denominator sinus power 2 end of power x power 3 end of power end of fraction dx = − fraction nominator 1 denominator 2 end of fraction cotangent x power 3 end of power + C, C is an element of the set R

Proposal 2

$$\int \frac{x^2}{\sin^2 x^3} dx = -\frac{1}{3} \cot x^3 + C \qquad C \in R \qquad (2)$$

Integral of fraction nominator x power 2 end of power denominator sinus power 2 end of power x power 3 end of power end of fraction dx = − fraction nominator 1 denominator 3 end of fraction cotangent x power 3 end of power + C, C is an element of the set R

Proposal 3

$$\int \frac{x^2}{\sin^2 x^3} dx = -\frac{1}{3} \tan x^3 + C \qquad C \in R \qquad (3)$$

Integral of fraction nominator x power 2 end of power denominator sinus power 2 end of power x power 3 end of power end of fraction dx = − fraction nominator 1 denominator 3 end of fraction tangent x power 3 end of power + C, C is an element of the set R

3 Results

Examples of screens corresponding to the solved exercise are presented in Figs. 5, 6 and 7.

The graphical interface of presented platform was designed and implemented as web application, using PHP and JavaScript technologies [8, 14]. Files of the exercise structure are generated once and are stored on server site, which allows the platform to be used simultaneously by many users. The user session provides possibility to archive an information about user activity.

The presented platform included about 240 prepared exercises from different areas of mathematics and 60 selected exercises including alternative description layer, created

Evaluate: $\int \frac{x^2 dx}{\sin^2 x^3}$

We integrate by substitution, which substitution do you select?

X $t = x^2$

◯ $t = x^3$

◯ $t = x$

◯   I want to remind yourself theorem of integration by substitution.

[Next]

Fig. 5 Example of a step of the solved exercise including selection list

Evaluate: $\int \dfrac{x^2 dx}{\sin^2 x^3}$

We can substitute $t = x^3$ and then we have:

$$\int \frac{x^2 dx}{\sin^2 x^3} = \begin{vmatrix} t = x^3 \\ dt = 3x^2 dx \\ \frac{1}{3} dt = x^2 dx \end{vmatrix} = \frac{1}{3} \int \frac{\boxed{}}{\sin^2 t}$$

Write the correct value in the text area.

Next

Fig. 6 Example of a step of the solved exercise including text area

according to proposed rules. About 1000 students and about 40 impaired students from 6 faculties of the university use this platform during math courses for both self and class learning. They used following components: Windows 7, 8,10 and Mac OS X operating systems, Mozilla Firefox, Internet Explorer, Chrome and Safari web browsers.

The accessibility and usability of developed e-learning platform was tested and evaluated by 23 impaired students: 12 low vision, 7 blind and 4 motor impaired, who had hand movement problems and used special keyboards or assistive devices. Blind students used the following popular screen reader: Jaws from Freedom Scientific Company [19], Windows Eyes from GW Micro Company [35] and Non Visual Desktop Access open source product [24]. Low vision students prefer using screen magnifiers like: Dolphin SuperNova Magnifier and Screen Reader from Dolphin Assistive Technology from life [11] or ZoomText MagReader from AiSquared Company [38].

The evaluation of the platform accessibility was conducted in few steps. The set of mathematical formulas and their alternative descriptions were prepared in the first step, which include dozens basic, complex and advanced formulas from different areas of mathematics. The evaluation criteria, which were collected in evaluation forms, were as follows:

- Recognition the structure of math formulas (the nominal scale: good, medium, poor; the students' comments describing difficulties),
- Simplicity of navigation in the structure of the equation (the nominal scale: good, medium, poor; the student's comments describing difficulties),

Evaluate: $\int \dfrac{x^2 dx}{\sin^2 x^3}$

Good answer, now we have:

$$\int \frac{x^2 dx}{\sin^2 x^3} = \begin{vmatrix} t = x^3 \\ dt = 3x^2 dx \\ \frac{1}{3} dt = x^2 dx \end{vmatrix} = \frac{1}{3} \int \frac{dt}{\sin^2 t} = -\frac{1}{3} \boxed{-?-}$$

$-?-$
$\cos t$
$\cot t$
$\tan t$

Select the correct answer from the drop-down

Next

Fig. 7 Example of a step of the solved exercise including drop-down selection list

- Understanding of math formulas (the nominal scale: good, medium, poor; the student's comments describing difficulties),

The blind students reported recognition level (75%) lower than low vision students (93%). The level of structure recognition was proportional to complexity of nested structures like fraction in fractions, several nested parenthesis, etc. Recognition of such structures required multiple navigation and reading aloud of substructure. The student emphasized the usability of additional words in alterative descriptions, containing information about the structure of the formulas.

As regard the second criteria, the scope of enable navigation modes resulted from available navigation modes of used screen readers and magnifiers: navigation by formulas, the elements of formulas (words), characters. Analysis of the results indicates that formulas and words navigation modes were the most frequently used. The character mode is useful when the long numbers exist in the formulas.

The study confirmed the good understanding of simple and complex formulas of 73–90% by blind and low vision students. The level of understanding of advanced formulas depends on the level of mathematical knowledge, what has been deduced based on the analysis of students' comments.

The second step of evaluation was focused on ergonomic and navigation through graphical user interface elements using keyboards or assistive technologies (the nominal scale: good, medium, poor; the students' comments describing difficulties). Ergonomic in this work includes consistency of size, arrangement, colour and naming of user interface elements. The 92% of users indicate good ergonomic of developed platform and full access and navigation using keyboards. At this stage of evaluation especially motor impaired students were engaged, who had hand movement problems.

At the last step of evaluation the efficiency of exercise solving by impaired student was performed. At the beginning, it was necessary to assess the math knowledge level from the particular math area of the participants. Each of them resolved three exercises of different level of complexity in traditional form and complete the math test. In the next step, the developed platform was used by students resolving exercise from the same area of mathematics. The complexity of this exercise was chosen individually for each student based on the result of the preliminary test. The assessment is based on the number of mistakes made, which are the sum of all elementary step during solving the particular kind of exercise at the specific topic and difficulty level. The mistakes made were available by user session logs. Based on the analysis of the most frequently mistakes made, system choose the next exercises related to the same issue and level of difficulties. If the student solved the exercise 100% correctly the platform proposes the next more complex exercise and the process is repeated.

Based on the analysis of users' logs (number of mistakes made, type and number of solving exercises) it can be concluded that:

- 12 out of 23 students improved the efficiency of exercise resolving - the number of mistakes made were decreased during the test.
- 6 out of 23 students also improved the efficiency of exercise resolving. The number of mistakes made were decreased but they had to resolve much more exercises of the same level of complexity comparing to the first group to achieve the similar level of efficiency.
- For 5 out of 23 students there were no progress during resolving math simple exercises, the number of mistakes made were at the high level. Moreover, based on the preliminary test

and students comments after the test in platform it can be conclude that their knowledge level of mathematics was inadequate to solving such type of exercises.

4 Discussion

The proposed method is the voice presentation of math formulas, containing additional information about their structure, understandable for visually impaired user. Introduced split into stages and interactive elements occurring at different stages enable a blind person to study math through self-solving exercises. The proposed method compared to the previously used, which is based on translations of conventional mathematical notation materials to Braille'a notation, has several advantages.

Preparing Braille'a materials containing math formulas is expensive and time-consuming for Braille transcribers. In addition, direct translation of text material may be insufficient due to the little explicit instructional information about representations [15]. Update and modification of prepared exercise are easier compared to Braille'a and DAISY standard. What is more, the collaboration between blind and teacher is easier and does not required knowledge of Braille 'a and Braille'a math extension notation from the teacher.

Preparing the concept of exercises and content in LATEX, including division, elaboration of assessment and transition conditions at every elementary step, consumes from 2 to 10 h. Based on the experience gained during the elaboration of an alternative description for 60 interactive exercises, it can be said that the time needed to prepare alternative descriptions of the exercise ranged from 2 to 5 h, depending on the number and complexity of elementary formulas. At the beginning of our research, the Speech Rule Engine was not in stable state. Moreover, it supports only English language [31]. This is the main reasons why we did not use it in our approach. Text descriptions used in prepared exercises were consulted with mathematicians. Reading aloud math formulas is different according to various culture, languages and user experience. Another important barrier in using Speech Rule Engine (SRE) was limited support for various screen readers and magnifiers. Other important reason is the fact that our solution uses LATEX notation to describe the mathematical formulas, which is not supported by the SRE. Supporting LATEX in SRE requires a translator between LATEX and MathML notations, which expands the stack of used technologies [29]. Some researchers raise the issue of ambiguity convert between LATEX and MathML notation [34]. Additionally, another report that TeX does not provide an encoding of the semantics for the mathematical material and this information must be derived from some (external) context [3]. Due to this fact, to guarantee high quality of education materials, we decided to avoid automatic conversion to alternative text description. In the future work automated translation of math formulas into native-language text will be considered. One important possible step, which significantly shorten the time for preparing the alternative description of formulas, will be the integration of acquired knowledge and experience to prepare the rules consistent with the XPath syntax, which can be integrate in SRE. Some of adapted exercises contain simple graphics which were described according to Graphics Description Guidelines developed by Carl and Ruth Shapiro Family National Center for Accessible Media at WGBH (NCAM) in conjunction with the DIAGRAM Center [9]. In order to develop an alternative chart description automatically, we are considering to use EvoGraphs from EvoXLabs Company [12].

During the further research and development it is intended to prepare math courses with interactive exercises from other areas of mathematics such as algebra and statistics. Moreover, the presented platform can be easily adapted to courses from other scientific and technical disciplines: chemistry, physics, informatics, etc.

References

1. Bernareggi C (2010) Non-sequential Mathematical Notations in the LAMBDA System. Computers helping people with special needs. ICCHP 2010. 12th International conference, Linz, Austria, July 14–16, 2010. Proceedings Springer. Berlin, 2010, LNCS 6180:385–39
2. Brzoza P, Łobos E, Acura J, Sikora B, Żabka M (2012) ForMath intelligent tutoring system in mathematics. In Proceedings of the 4th International Conference on Computer Supported Education, CSEDU 2012, Porto. 118–122. doi:10.5220/0003899901180122
3. Brzoza P, Maćkowski M (2014) Intelligent Tutoring Math Platform Accessible for Visually Impaired People. Computers Helping People with Special Needs: 14th International Conference, ICCHP 2014, Paris, France, July 9–11, 2014, Proceedings, Part I, 2014. 519–524. doi:10.1007/978-3-319-08596-8_81
4. Brzoza P, Spinczyk D (2006) Multimedia browser for Internet Online Daisy books, Computers Helping People with Special Needs: 10th International Conference, ICCHP 2006, Linz, Austria, July 11–13, Proceedings, Part I, Springer Berlin Heidelberg, Book Series: Lecture Notes In Computer Science 4061:1087–1093
5. Cervone D (2012) Math Jax: a platform for mathematics on the web (PDF). Not Am Math Soc 59(2):312–316. doi:10.1090/noti794
6. Cooper M, Lowe T, Taylor M (2008) Access to Mathematics in Web Resources for People with a Visual Impairment. Computers Helping People with Special Needs: 11th International Conference, ICCHP 2008, Linz, Austria, July 9–11, Proceedings, Springer Berlin Heidelberg, 926—933
7. Craiga S, Hub X, Graesserb A, Bargagliottic A, Sterbinskyb A, Cheneyb K, Okwumabuab T (2013) The impact of a technology-based mathematics after-school program using ALEKS on student's knowledge and behaviors. Comput Educ 68:495–504
8. Cui W, Huang L, Liang L, Li J (2009) The research of PHP development framework based on MVC pattern, Computer Sciences and Convergence Information Technology, 2009. ICCIT '09. Fourth International Conference on, Seoul, 947–949. doi:10.1109/ICCIT.2009.130
9. Diagram center – website: http://diagramcenter.org/. Accessed 21 September 2016
10. Dolphin Easy Reader: http://www.yourdolphin.com/productdetail.asp?id=9. Accessed 21 September 2016
11. Dolphin SuperNova Magnifier and Screen Reader - manufacturer's website: https://yourdolphin.com/supernova-magnifier-screen-reader. Accessed 21 September 2016
12. Evographs - manufacturer's website: http://evoxlabs.org/projects/evographs. Accessed 21 September 2016
13. Fateman R (2013) How can we speak math? Unpublished manuscript. https://people.eecs.berkeley.edu/~fateman/papers/speakmath.pdf. Accessed 21 September 2016
14. Gamrat B (2006) PHP and preprocessed Web. Dr Dobb'S J 31(1):46–48
15. Garderen D, Scheuermann A, Jackson C (2012) Developing representational ability in mathematics for students with learning disabilities a content analysis of grades 6 and 7 textbooks. Learn Disabil Q 35(1):24–38. doi:10.1177/0731948711429726
16. Gardner J A (2014) The LEAN Math Accessible MathML Editor, Computers Helping People with Special Needs: 14th International Conference, ICCHP 2014, Paris, France, July 9–11, 2014, Proceedings, Part I. Springer International Publishing, 580–587
17. Goguadze G, Gonźalez Palomo A, Melis E Interactivity of Exercises in ActiveMath, IOS Press, 2005. Link to the tool: http://www.activemath.org/

18. Henderson S (2014) MiniMatecaVox: math teaching application aimed to visually impaired children in literacy phase. Bibliotheca Digital da Unicamp. Link to the tool: http://imasters.expert/minimatecavox-helps-visually-impaired-children-learn-mathematics/. Accessed 21 September 2016

19. Job Access With Speech - manufacturer's website: http://www.freedomscientific.com/Products/Blindness/JAWS. Accessed 21 September 2016

20. Kerscher G (2001) DAISY Consortium: information technology for the world's blind and print-disabled population – past, present, and into the future. Library Hi Tech 19(1):11–15

21. Lamport L (1994) LaTeX: A Document Preparation System. Addison-Wesley Professional

22. Namahoe K (2014) Khan Academy Unveils New Math Resources for Common Core. The Journal. https://thejournal.com/articles/2014/03/24/khan-academy-unveils-new-math-resources-for-common-core.aspx. Accessed 21 September 2016

23. Nicotra G, Bertoni G, Bortolazzi E, Formenti L (2010) Innovative Methods for Accessing Mathematics by Visually Impaired Users. Computers helping people with special needs. ICCHP 2010. 12th International conference, Linz, Austria, July 14–16, 2010. Proceedings Springer. Berlin, 2010, LNCS 6180:423–430

24. Non visual Desktop Access - repository website: https://github.com/nvaccess/nvda. Accessed 21 September 2016

25. Riga P, Kouroupetroglou G, Ioannidou P (2016) An Evaluation Methodology of Math-to-Speech in Non-English DAISY Digital Talking Books BT -Computers Helping People with Special Needs: 15th International Conference, ICCHP 2016, Linz, Austria, July 13–15, Proceedings, Part I, Springer Berlin Heidelberg, 27–34

26. Salamonczyk A, Brzostek-Pawlowska J (2015) Translation of MathML formulas to Polish text, example applications in teaching the blind. Cybernetics (CYBCONF), 2015 I.E. 2nd International Conference on, Gdynia. 240–244. doi:10.1109/CYBConf.2015.7175939

27. Soiffer N (2009) A Flexible Design for Accessible Spoken Math, Universal Access in Human-Computer Interaction. Applications and Services: 5th International Conference, UAHCI 2009, Held as Part of HCI International 2009, San Diego, CA, USA, July 19–24, 2009. Proceedings, Part III, Springer Berlin Heidelberg, 130–139

28. Soiffer N (2016) A Study of Speech Versus Braille and Large Print of Mathematical Expressions, Computers Helping People with Special Needs: 15th International Conference, ICCHP 2016, Linz, Austria, July 13–15, Proceedings, Part I, Springer Berlin Heidelberg, 59–66

29. Sorge V (2016) Supporting Visual Impaired Learners in Editing Mathematics. Proceeding ASSETS '16 Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility, 323–324, Reno. doi:10.1145/2982142.2982212

30. Sorge V Speech Rule Engine. https://github.com/zorkow/speech-rule-engine. Accessed 21 September 2016

31. Sorge V, Chen Ch, Raman T, Tseng D (2014) Towards making mathematics a first class citizen in general screen readers. Proceeding W4A '14 Proceedings of the 11th Web for All Conference Article No. 40. Seoul. doi:10.1145/2596695.2596700

32. Spinczyk D, Brzoza P (2008) Multimedia system for accessible distant education. Info Technol Biomed 47:513–517

33. Unicode Technical Report #25: Unicode Support for Mathematics" (2015). The Unicode Consortium. 31 July 2015: http://www.unicode.org/reports/tr25/. Accessed 21 September 2016

34. Wang K, Li X, Tian X (2015) On Ambiguity Issues of Converting LaTeX Mathematical Formula to Content MathML. Collaborative Computing: Networking, Applications, and Work sharing: 11th International Conference, CollaborateCom 2015, Wuhan, November 10–11, 2015, China. Proceedings. Springer International Publishing. 289–295. doi:10.1007/978-3-319-28910-6_26

35. Window-Eyes - manufacturer's website: http://www.gwmicro.com/window-eyes/. Accessed 21 September 2016

36. Yamaguchi K, Suzuki M (2012) Accessible Authoring Tool for DAISY Ranging from Mathematics to Others. Computers Helping People with Special Needs: 13th International Conference, ICCHP 2012, Linz, Austria, July 11–13, Proceedings, Part I, Springer Berlin Heidelberg, 130–137

37. Żabka M (2016) The use of LaTeX in building of the educational platform ForMath. In Proceedings of BachoTeX 2016 Conference, (to publish). http://www.gust.org.pl/bachotex/2016-en/program#section-121. Accessed 21 September 2016

38. ZoomText MagReader - manufacturer's website: https://www.zoomtext.com/products/zoomtext-magnifierreader/. Accessed 21 September 2016

Michał Maćkowski is a graduate at the Faculty of Automatic Control, Electronics and Computer Science. Currently employee at the Computer Science Department of Silesian University of Technology. His scientific interest include the wide range of multimedia and text-based information retrieval, multimedia in web-applications, computer networking, programming of mobile devices.

Piotr Brzoza is a graduate at the Faculty of Automatic Control, Electronics and Computer Science. Currently employee at the Computer Science Department of Silesian University of Technology. Research interests: assistive technology, multimedia systems, e-learning.

Marek Żabka is graduate of mathematics at the University of Silesia. Currently employee at the Faculty of Applied Mathematics. Research interests: mathematical algebra, latex, multimedia, web applications.

Dominik Spinczyk associate professor at the Faculty of Biomedical Engineering Silesian University of Technology. His scientific interests include processing of multimedia data, in particular for computer-aided diagnosis and therapy.