Implementation of a Model-Tracing-Based Learning Diagnosis System to Promote Elementary Students’ Learning in Mathematics

Yian-Shu Chu¹, Haw-Ching Yang², Shian-Shyong Tseng³ and Che-Ching Yang¹

¹Department of Computer Science, National Chiao Tung University, Hsinchu, 30010, Taiwan // ²Institute of System Information and Control, National Kaohsiung First University of Science and Technology, Kaohsiung, 811, Taiwan // ³Department of Applied Informatics and Multimedia, Asia University, Taichung, 41354, Taiwan // 

*Corresponding author

(Submitted July 12, 2012; Revised December 13, 2012; Accepted March 16, 2013)

ABSTRACT

Of all teaching methods, one-to-one human tutoring is the most powerful method for promoting learning. To achieve this aim and reduce teaching load, researchers developed intelligent tutoring systems (ITSs) to employ one-to-one tutoring (Aleven, McLaren, & Sewall, 2009; Aleven, McLaren, Sewall, & Koedinger, 2009; Anderson, Corbett, Koedinger, & Pelletier, 1995; Anderson & Reiser, 1985; Blessing, Gilbert, Ourada, & Ritter, 2009; Mitrovic et al., 2009; Mitrovic & Ohlsson, 1999; Suraweera, Mitrovic, & Martin, 2007; VanLehn et al., 2005). Most ITSs have restricted user interfaces, which confine reasoning strategies of students during problem solving, thus ignoring the fact that students could use dissimilar strategies to solve a given question. Furthermore, student learning problems could be diagnosed from the derivation of their answers. In order to interpret students’ mathematical problem-solving behaviors, this study developed a Model-tracing Intelligent Tutor (MIT) to diagnose students’ learning problems and provide learning feedback for individual students. A quasi-experiment was conducted in an elementary school to evaluate the effectiveness of the proposed approach, in which 124 fifth graders participated. The experimental results show that the model-tracing-based learning diagnosis system is significantly more helpful to the students in learning mathematics than the conventional web-based test in terms of learning achievements.

Keywords
Learning diagnosis, Computer-assisted learning, Computer-assisted testing, Model tracing

Introduction

One-to-one human tutoring, which has been shown to be much more effective than conventional classroom instruction, enables higher achievements in most students (Bloom, 1984; Chou, Huang, & Lin, 2011; VanLehn, 2006). However, one-to-one human tutoring is extremely costly, and one-to-many classroom instruction leaves little time for teachers to take care of individual student needs (Zinn & Scheuer, 2006). Intelligent tutoring systems (ITSs) have been developed to cope with the problems stated above and to employ one-to-one tutoring (Aleven, McLaren, & Sewall, 2009; Aleven, McLaren, Sewall, et al., 2009; Anderson et al., 1995; Anderson & Reiser, 1985; Blessing et al., 2009; Mitrovic et al., 2009; Mitrovic & Ohlsson, 1999; Suraweera et al., 2007; VanLehn et al., 2005).

Although researchers have demonstrated the benefits of ITSs in a range of domains (Chou et al., 2011; Chu, Hwang, & Huang, 2010; Lee & Bull, 2008; VanLehn et al., 2005), some issues still need to be further discussed (VanLehn et al., 2005). One of the issues is many ITSs have user interfaces that guide students’ reasoning strategies by restricting the intermediate steps that students should follow. In other words, those ITSs adopted the restricted user interfaces to confine reasoning by offering a fixed type-in box for entering the intermediate steps. Because a given question can be solved by different strategies, students’ reasoning process should not be confined.

Singley and Anderson (1989) pointed out that one gets higher transfer from training to testing when the user interfaces are similar. Therefore, students who use a tutoring system with a less constrained user interface to learn might have similar learning gains to those who use pencil and paper (VanLehn et al., 2005). Moreover, keeping the user interface less constrained makes the tutoring system less invasive.

In addition, in a paper-and-pencil calculation test, learning problems of students could be diagnosed from answering derivations (in particular, the step-by-step reasoning processes). The answering derivations denote the necessary expressions, equations, or functions while solving a mathematical problem. Because students may have different
levels of misconceptions or mistakes during the problem-solving process, development of a less invasive tutoring system that can friendly interpret mathematical problem-solving behaviors is an important issue.

In general, there are three types of ITSs (cognitive tutors, constraint-based tutors, and example-tracing tutors), which possess different degrees of machine intelligence and provide different interaction mechanisms and different implementation complexities. Cognitive tutors apply a model-tracing approach to interpret and assess student behavior with reference to a cognitive model (Anderson et al., 1995; Anderson & Reiser, 1985; Blessing et al., 2009; VanLehn et al., 2005). Constraint-based tutors apply a constraint-based modeling approach to interpret and assess student work with respect to a set of constraints (Mitrovic, Martin, Suraweera, Zakharov, Milik, & Holland, 2009; Mitrovic, & Ohlsson, 1999; Suraweera, Mitrovic, & Martin, 2007). Example-tracing tutors apply the example-tracing approach to interpret and assess student behavior with reference to generalized examples of problem-solving behavior (Aleven, McLaren, & Sewall, 2009; Aleven, McLaren, Sewall, et al., 2009).

Since the example-tracing tutor’s knowledge is not generalizable to other similar problems, it fails to reason multiple problem scenarios by itself. This research is to discover the strategies and misconceptions students may acquire; therefore, a model-tracing-based approach is used to compare students’ activities with a cognitive model of student problem solving to achieve the goal of interpretation of students’ behaviors. Previous research, such as LISP Tutor (Anderson & Reiser, 1985), algebra cognitive tutor (Anderson et al., 1995), and Andes (VanLehn et al., 2005) showed that the model-tracing approach not only can analyze cognitive behaviors of students but can also evaluate their knowledge.

In addition, as a minimum requirement an ITS must have an “inner loop,” that is, provide minimal feedback within problem-solving activities (VanLehn, 2006). In this study, minimal feedback is to provide a timely diagnosis report on the final result (whether the whole solution is correct) and the working steps (which step is incorrect and what is the cause of error).

In order to cope with previous problems, this study has developed a testing and diagnostic system based on tutoring behavior identified by VanLehn (2006). The proposed system, Model-tracing Intelligent Tutor (MIT), includes four components: (1) lexical analyzer (scanner); (2) syntax analyzer (parser); (3) semantic analyzer; and (4) report generator. MIT is implemented with the aim of conducting a one-to-one tutoring mechanism with instant feedback to improve learning in mathematics of students. Therefore, the research question is “what are the learning achievements of students after using MIT.” Finally, an experiment on a fraction lesson in a mathematics course was conducted to demonstrate the effectiveness of the proposed system.

The remainder of this paper is organized as follows. Section 2 describes student problem-solving behavior in fractions. Section 3 presents how we built MIT and how MIT can be used as a web-based test system. Section 4 describes the methodology to evaluate the presented MIT and experimental results. Section 5 presents discussions and future research directions, and Section 6 draws conclusions.

Student problem-solving behavior in fractions

Following the work of Aleven, McLaren, Sewall, et al. (2009), solutions for a mathematical problem could be represented as a behavior graph (directed and acyclic), which may contain multiple paths corresponding to different ways of solving the problem. For example, a problem-solving behavior to solve a fraction question such as “adding mixed fractions” could be as follows: converting a mixed number to an improper fraction, reducing fractions to a common denominator, adding fractions with common denominators, and converting an improper fraction to a mixed number. Another way of solving the problem is by adding integers, reducing fractions to a common denominator, adding fractions with common denominators, and converting an improper fraction to a mixed number.

Also, a behavior graph may contain incorrect behavior links. In the graph, the links represent problem-solving actions, and the nodes show problem-solving states. In terms of incorrect behavior, when solving fractions, students may have common misconceptions that lead to errors in computation (Idris & Narayanan, 2011; Lee & Bull, 2008; Stead, 2012; Tatsuoka, 1984; Tirosh, 2000). Because of rote memorization and insufficient knowledge, students may overuse generalize rules and procedures when following the steps in worked-out examples (Idris & Narayanan,
2011). Therefore, errors of students are often systematic and rule-based rather than random (Idris & Narayanan, 2011). Table 1 illustrates the typical misconceptions students may have.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Behavior description</th>
<th>Example</th>
<th>No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mixed to improper fractions</td>
<td>Move the whole number to the numerator</td>
<td>(\frac{3}{5} = \frac{13}{5})</td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>Add the whole number to the numerator</td>
<td>(\frac{3}{5} = \frac{4}{5})</td>
<td>B</td>
</tr>
<tr>
<td>Raising a fraction to higher terms</td>
<td>Multiply the numerator and the denominator by two different numbers, respectively</td>
<td>(\frac{2}{5} = \frac{10}{20})</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>A random number is chosen and that number is added to both the numerator and the denominator</td>
<td>(\frac{2}{5} = \frac{4}{7})</td>
<td>D</td>
</tr>
<tr>
<td>Reducing a fraction to lower terms</td>
<td>Divide the numerator and the denominator by two different numbers, respectively</td>
<td>(\frac{4}{8} = \frac{1}{4})</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>A random number is chosen and that number is subtracted from both the numerator and the denominator</td>
<td>(\frac{2}{10} = \frac{1}{9})</td>
<td>F</td>
</tr>
<tr>
<td>Adding fractions with uncommon denominators</td>
<td>Multiply the numerators together and multiply the denominators together</td>
<td>(\frac{2}{5} + \frac{1}{2} = \frac{2}{10})</td>
<td>G</td>
</tr>
<tr>
<td></td>
<td>Add the numerators and multiply the denominators</td>
<td>(\frac{2}{5} + \frac{1}{2} = \frac{3}{10})</td>
<td>H</td>
</tr>
<tr>
<td></td>
<td>Add the numerators together and add the denominators together</td>
<td>(\frac{2}{5} + \frac{1}{2} = \frac{3}{7})</td>
<td>I</td>
</tr>
</tbody>
</table>

Model-tracing Intelligent Tutor (MIT)

A fraction question may comprise several sub-questions which may be solved in a variety of orders or ways. Furthermore, students may have misconceptions that lead to errors in computation. To interpret and assess student behavior, a learning diagnosis system, “Model-tracing Intelligent Tutor (MIT)” was developed to trace mathematical step-by-step operations of students after answering a fraction question. The running phase of MIT is as shown in Figure 1, which includes four components: lexical analyzer (scanner), syntax analyzer (parser), semantic analyzer, and report generator. The lexical analyzer reads the input stream (e.g., mathematical equation) and passes tokens to the parser. The syntax analyzer reads the tokens and identifies the syntactic structure. The semantic analyzer drives semantic processing. Finally, the report generator generates diagnosis results for students.
In this study, Lex, the lexical analyzer generator, is used to generate a scanner for dealing with lexical analysis. Yet Another Compiler Compiler (YACC) (Johnson, 1975) is used to generate a set of parse tables for analyzing syntax and semantic. To specify the effectiveness of the model-tracing mechanism, we address the use of YACC specifications to generate this learning diagnosis system. Also, at the end of this section, we describe the development of MIT.

The specification

YACC, a context-free language parser generator, is a look-ahead left-to-right, rightmost-derivation (LALR) parser generator developed by AT&T Bell Laboratory in C language, which is used to generate parsers with a given input file. YACC also supports a general mechanism for semantic analysis. The input file consists of three sections: declarations, productions, and subroutines. To develop the MIT system, the declarations section defines mathematical symbols used during operations, the productions section specifies rules for mathematical equations, and the subroutines section denotes rules for correctness determination of solving fractions. Below are the related definitions and examples.

Definition 1 (declarations). Mathematical symbols (e.g., operands and operators) used in an equation are defined in the declarations section.

An expression is a set of terms with mathematical operations combining them, and an equation consists of two expressions with a relational symbol between them. Some terms related to an equation are listed below.

- **Token**: There are nine different tokens used in an equation: “INT” (integer), “/” (fraction bar), “+” (addition), “-” (subtraction), “*” (multiplication), “/” (division), “(” (left parentheses), “)” (right parentheses), and “=” (equal).
- **Primary**: A fraction is called a primary in the specification, which is written in the form \( \frac{c}{b} \), where \( a, b \) and \( c \) are integers and \( b \) cannot be 0. The number \( c \) is called the numerator, and the number \( b \) is called the denominator. There are three kinds of primaries: proper and improper common fractions and mixed numbers. The fraction is called proper if the numerator is less than the denominator, and improper otherwise. A mixed number consists of an integer and a proper fraction.
- **Operator**: The Operators are “+”, “-”, “*”, “/”, “(”, “)”, and “=”.

Definition 2 (productions). Rules of the mathematical equation expressions are defined in the productions section.

Four rules are listed below. Eqs. 1 to 3 show the productions of fraction addition. Eq. 4 shows a primary can be recognized if its format is \( \frac{a}{b} \).

\[
\begin{align*}
\text{<expression>} & \rightarrow \text{<expression>} + \text{<multiplicative expression>} \\
\text{<expression>} & \rightarrow \text{<multiplicative expression>} \\
\text{<multiplicative expression>} & \rightarrow \text{<primary>} \\
\text{<primary>} & \rightarrow \frac{\text{INT}}{\text{INT}}
\end{align*}
\]

(1) (2) (3) (4)

Example 1 (syntax analysis):

According to Eq. 4, the expression \( \frac{3}{5} + \frac{1}{2} \) can be recognized as two primaries with one operator combining them. Then, this expression can be recognized as an expression of fraction addition by using Eqs. 1 to 3.
**Definition 3 (subroutines).** The subroutines in YACC, which are rules for correctness determination of solving fraction, are used to analyze learning status of students while solving fraction questions.

Three rules are listed below for converting a mixed number to an improper fraction. Among the rules, Eq. 5 is a correct pattern. This pattern is used to recognize correct conversion of a mixed number to an improper fraction, which consists of two intersection rules: \( Y_2 = X_2, Y_3 = X_3 + X_2 \times X_1 \). Since students may have misconceptions while solving questions, Eq. 6 and Eq. 7 are illustrative misconceptions, A and B, respectively, which are mentioned in Table 1 (Idris & Narayanan, 2011; Lee & Bull, 2008; Stead, 2012; Tatsuoka, 1984; Tirosh, 2000).

\[
\frac{X_1}{X_2} = \frac{Y_3}{Y_2}
\]

Correct pattern \( Y_2 = X_2 \cap Y_3 = X_3 + (X_2 \times X_1) \) \( \quad (5) \)

Misconception A \( Y_2 = X_2 \cap Y_3 = X_3 + (10 \times X_1) \) \( \quad (6) \)

Misconception B \( Y_2 = X_2 \cap Y_3 = X_3 + X_1 \) \( \quad (7) \)

**Example 2 (correct behavior):**
Following the rule of Eq. 5, an equation \( \frac{3}{5} + \frac{8}{5} \) can be recognized as a fraction question solving step of converting a mixed number to an improper fraction during the fraction solving process.

**Example 3 (incorrect behavior):**
By using Eq. 6, an equation \( \frac{3}{5} = \frac{13}{5} \) can be recognized as an error step of converting a mixed number to an improper fraction. Similarly, an equation \( \frac{3}{5} = \frac{4}{5} \) also can be recognized as an error step with the aid of Eq. 7.

**The development**
We implemented the leaning diagnosis system MIT based on the specifications. Figure 2 shows the left-to-right trace process of MIT. Three different kinds of operations are adopted in this figure: raising a fraction to higher terms, adding fractions with common denominators, and converting an improper fraction to a mixed number.

![Figure 2. The trace process](image)

This system characterizes not only strategies but also the misconceptions that students may acquire by mapping students’ problem-solving steps to error steps. Also, MIT gives immediate feedback right after students click on the Submit Answer button, i.e., feedback is given after the completion of every item. Then the tutor analyzes the entered answers of all the steps and determines the giving diagnosis results.

As shown in Figure 3, students key their responses into a text box. To display mathematical expressions in web browsers, Mathematical Markup Language (MathML, an XML-based markup language recommended by the W3C math working group [http://www.w3.org/Math/]), is used to describe mathematical notations and capture both its
structure and content. Table 2 illustrates the MathML’s representation of \((a + b)^2\). Because editing MathML directly to express mathematical expressions is difficult for students, a client-side mathematical markup language ASCIIMathML (http://www1.chapman.edu/~jipsen/mathml/asciimath.html) is used in this study to assist students in typing mathematical expressions with ASCII codes and converting the expressions to MathML syntax. In other words, students type a fraction “6 4/5” and browsers will display the typed text as \(6 \frac{4}{5}\).

As shown in Figure 3, students key their responses into a text box. To display mathematical expressions in web browsers, Mathematical Markup Language (MathML, an XML-based markup language recommended by the W3C math working group http://www.w3.org/Math/), is used to describe mathematical notations and capture both its structure and content. Table 2 illustrates the MathML’s representation of \((a + b)^2\). Because editing MathML directly to express mathematical expressions is difficult for students, a client-side mathematical markup language ASCIIMathML (http://www1.chapman.edu/~jipsen/mathml/asciimath.html) is used in this study to assist students in typing mathematical expressions with ASCII codes and converting the expressions to MathML syntax. In other words, students type a fraction “6 4/5” and browsers will display the typed text as \(6 \frac{4}{5}\).

Meanwhile, an input assistant tool is provided as another input method, as shown in the bottom of Figure 3. Students can choose to type fractions or mathematical notations to input their answers via the input assistant tool. For example, students input a fraction and then click the operator “add” to display it in the text box. If students enter an incorrect equation, a pop-up window will prompt to display the error message “Please check your equation carefully!”

Table 2. The MathML representation of \((a + b)^2\)

```
<msup>
  <mfenced>
    <mi>a</mi>
    <mo>+</mo>
    <mi>b</mi>
  </mfenced>
  <mn>2</mn>
</msup>
```
After students finish answering the questions, the MIT system starts to trace their operations. MIT traces not only the final result but also the operating steps. The step analyzer shows the operation result as well as the cause of error.

Figure 4 shows the diagnosis results of operations of the student in Figure 3. The student used two operations: adding fractions with uncommon denominators and reducing a fraction to lowest terms. Because of the incorrect operation of “adding fractions with uncommon denominators,” the final result is incorrect. The cause of error in this operation is “add the top numbers and the bottom numbers.”

![Figure 4. A screenshot of learning diagnosis](image)

**The experiment and evaluation**

In order to evaluate the effectiveness of this approach, we conducted a three-week experiment. This research adopted a quasi-experimental design. There were 124 fifth-grade students participating in this research. The students were from four classes and taught by the same instructor under the same conditions. They had previously taken computer courses and possessed basic computer skills. The four participating classes were randomly divided into experimental group and control group, each of which consisted of sixty-two students. For the students in the experimental group, the answers of individual students were analyzed by using MIT. Each student in the experimental group can get a detailed analysis of incorrectly answered item along with the test results. The students in the control group used a conventional web-based test system and only the test results (correct or incorrect) were given. The only difference between a conventional web-based test system and a traditional paper-and-pencil test is that a conventional web-based test system can be done online.

During the three weeks, the two groups of students received instruction about fraction calculations; moreover, a paper-and-pencil pre-test was conducted to analyze the students’ knowledge. Then, the students in two groups were instructed with the tools of the learning activity. After the instruction, we conducted a 60-minute learning activity. All students participated in web-based test system. Students in the experimental group used MIT while students in the control group used conventional web-based test system. After the learning activity, all participants took a paper-and-pencil post-test.

Before the experiment began, three senior mathematics teachers who had more than five years of experience in teaching mathematics selected thirty fractional additions as item candidates. To calculate item difficulty and item discrimination, teacher participants administered these items to the students in four other different classes at the same elementary school. According to difficulty and discrimination, we chose ten items for pre-test and other similar ten items for post-test. The average difficulty of the pre-test and post-test was 0.539 and 0.532, respectively. The discrimination index of each item in both tests was over 0.500.
The scores of the students in the pre- and post-tests were analyzed to compare the learning achievements of the students in the two groups. Table 3 shows the independent t-test of the pre-test scores between the two groups. In the pre-test, the average scores of the two groups were 55.48 and 60.00, and the control group had a higher average score than the experimental group. The t-test ($t = 0.826, p > .05$) shows that there is no difference between the two groups’ average pre-test scores. In other words, students in both groups had the same prior knowledge of fractions before the implementation of the experimental treatments.

<table>
<thead>
<tr>
<th>Groups</th>
<th>N</th>
<th>Mean</th>
<th>SD</th>
<th>t-test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experimental</td>
<td>62</td>
<td>55.48</td>
<td>28.61</td>
<td>0.826</td>
</tr>
<tr>
<td>Control</td>
<td>62</td>
<td>60.00</td>
<td>32.14</td>
<td></td>
</tr>
</tbody>
</table>

Table 4 shows the ANCOVA result on the post-test scores of the two groups, the means and standard deviations of the post-test scores, which were 69.68 and 30.00 for the experimental group and 56.61 and 34.39 for the control group. It was found that the post-test scores of two groups were significantly different, with $F = 20.27 (p < .001)$; implying that the students in the experimental group achieved a better learning performance in the knowledge of fractions after using MIT than those using a conventional web-based test system. Moreover, the adjusted mean of the experimental group's post-test scores (71.55) was higher than that of the control group (54.74). The paired t-test result of the control group ($t = 1.126, p > .05$) indicates that the mean difference between the pre-test and post-test measures of the control group of students is not significantly different. Consequently, we conclude that the model-tracing intelligent tutor seems to be more effective than a conventional web-based test system in promoting the learning achievements of the students.

<table>
<thead>
<tr>
<th>Groups</th>
<th>N</th>
<th>Mean</th>
<th>SD</th>
<th>Adjusted Mean</th>
<th>SE</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experimental</td>
<td>62</td>
<td>69.68</td>
<td>30.00</td>
<td>71.55</td>
<td>2.64</td>
<td>20.27***</td>
</tr>
<tr>
<td>Control</td>
<td>62</td>
<td>56.61</td>
<td>34.39</td>
<td>54.74</td>
<td>2.64</td>
<td></td>
</tr>
</tbody>
</table>

* $p < .05$. ** $p < .01$. *** $p < .001$.

**Discussions**

The provision of feedback is believed to lead to better learning outcomes (Hattie & Timperley, 2007; Kleij, Eggen, Timmerse, & Veldkamp, 2012; Wu, Hwang, Milrad, Ke, & Huang, 2012). Timely feedback increases motivation and tends to motivate students to engage in learning (Kleij et al., 2012). Also, feedback can fill a gap between what is understood and what is aimed to be understood (Hattie & Timperley, 2007; Sadler, 1989). However, the existing web-based assessment systems mainly provide feedback based on multiple-choice question type or a restricted user interface. Through the proposed one-to-one tutoring mechanism, the instant feedback is provided to students. This system can discover students’ errors or misconceptions from their answers with model tracing approach, not only on the final results, but also on every working step.

As a result, the experimental results show that the model-tracing intelligent tutor helps students achieve significantly better effectiveness in improving students’ learning achievements than using the conventional web-based test system. That is, the proposed learning diagnosis system discovers students’ current knowledge and their weaknesses. Then students are able to make up lost ground and to improve their learning achievements. Namely, the proposed system, MIT, helps students to develop a deeper understanding of fractions. Besides, via this system, each student’s misconceptions are recorded; thus, teachers can easily understand students’ weaknesses and then help them to learn.
better, faster, and more easily. In addition, it should be noted that the applications of MIT could be expanded to science, engineering, and mathematics courses that require solving equation problems.

Although this approach seems to be promising, it has some limitations in its practical application. For example, in order to trace students’ behaviors precisely and diagnose what kind of problems students have exactly, the question solving process need to be performed by typing answers into the text box step-by-step. Another limitation is that one misconception could be caused by dissimilar reasons (e.g., insufficient prerequisite knowledge). According to the researches (Chu et al., 2010; Hwang, 2003), there are prerequisite relationships among the concepts in a course. Since a lack of different prerequisite knowledge can lead to learning difficulties, future work of MIT will be directed to investigate how learning guidance could be provided for individual students by finding a set of relevant poorly learned concepts.

Conclusions

Manipulating fractions is considered one of the fundamental mathematical skills learned in elementary school. Therefore, learning about fractions has an important role in the course of children’s mathematical study (Armstrong & Larson, 1995; Barash & Klein, 1996; Behr, Harel, Post, & Lesh, 1992; Behr & Post, 1992; Behr, Wachsmuth, & Post, 1985; Hunting, 1983). However, fractions are difficult for elementary school students. Students often solve fraction problems by relying on the strategy of searching through their memories for a previously taught algorithm without understanding of the fundamental nature of fractions (Kerslake, 1986). Learning by rote memorization, however, may cause students to have misconceptions. Researchers indicated that finding students’ misconceptions or bugs is helpful for teachers and students (Brown & Burton, 1978). Schwarzenberger (1984) pointed out that mistakes help us to realize current learning situations. Also, mistakes can aid the process of mathematical discovery and assist mathematical understanding, and they can tell us more about what might be happening in a pupil’s mind than any number of correct answers (Schwarzenberger, 1984). In addition, Kerslake (1986) suggested immediate feedback could reduce students’ fear of fractions.

To assist learners in learning fractions, various cognitive tools have been devised to support, guide, and mediate the cognitive processes of learners, and meet the diverse needs of learners in comprehending procedural knowledge (Kong, 2008; Kong & Kwok, 2005). Lee and Bull (2008) introduced a learning environment with an open learner model, which models learners’ current understanding of the domain. Its aim was to help children understand their problems with fractions and help parents to help their children overcome misconceptions. To further diagnose students’ learning problems, researchers have demonstrated the benefits of applying learning diagnosis mechanisms in various courses (Hwang, 2003; Hwang, Tseng, & Hwang, 2008).

In this paper, an innovative learning diagnosis system, Model-tracing Intelligent Tutor (MIT), has been proposed to assist teachers to diagnose student learning problems. MIT is built on a context-free language parser generator YACC with input file (declarations, productions, and subroutines). By using MIT, students’ mathematical operations can be precisely traced step-by-step. When an item is incorrectly answered, MIT provides a timely diagnostic report. With the aid of this computer-assisted approach, teachers not only understand students’ learning status (diagnosis results) but also save time; therefore, teachers are able to assist weak pupils with remedial instructions.

This research adopts a quasi-experimental design to investigate the effectiveness of this system and a conventional web-based test. The research findings show that students using MIT achieve better learning than those using a conventional web-based test.

Acknowledgments

This work was partially supported by the National Science Council of the Republic of China under Grant No. NSC 98-2511-S-468-004-MY3, NSC 99-2221-E-009-130-MY2, NSC 100-2511-S-468 -002, and NSC 101-2511-S-468 -007 -MY3. The authors would like to thank Barbara Adamski for valuable comments to revise this paper.
References


