Computer-assisted learning for mathematical problem solving

Kuo-En Chang a,*, Yao-Ting Sung b, Shiu-Feng Lin a

a Department of Information and Computer Education, National Taiwan Normal University, 162, Ho Ping East Road, Sec 1, Taipei, Taiwan, ROC
b Department of Educational Psychology and Counseling, National Taiwan Normal University, Taipei, Taiwan, ROC

Abstract

Previous computer-assisted problem-solving systems have incorporated all the problem-solving steps within a single stage, making it difficult to diagnose stages at which errors occurred when a student encounters difficulties, and imposing a too-high cognitive load on students in their problem solving. This study proposes a computer-assisted system named MathCAL, whose design is based on four problem-solving stages: (1) understanding the problem, (2) making a plan, (3) executing the plan and (4) reviewing the solution. A sample of one hundred and thirty fifth-grade students (aged 11 years old) completed a range of elementary school mathematical problems and empirically demonstrated. The results showed MathCAL to be effective in improving the performance of students with lower problem solving ability. This evaluation allowed us to address the problem of whether the assistances in various stages help students with their problem solving. These assistances improve students’ problem-solving skills in each stage.

Keywords: Elementary education; Interactive learning environment; Teaching/learning strategies

1. Introduction

In order to help students better cope with difficulties encountered in solving problems, many researchers have developed computer-assisted mathematical problem-solving tools based on...
visualization strategies (Mayer, 1992; Silver, 1987). For example, the computer-assisted problem-solving systems developed by Derry and Hawkes (1993) and Reusser (1996) adopt schema representations and solution trees. Students can conceptualize many mathematical problems through a schema, and then describe the solution steps in detail by constructing a solution tree. The solution tree can help students understand and solve complicated mathematical problems. However, the following problems presented in existing computer-assisted problem-solving systems warrant further investigation:

1. Previous computer-assisted problem-solving systems covered the entire procedure of problem solving, reading the problem, planning, calculation and verification, in a single stage. As a result, the systems cannot be used to diagnose precisely what stage(s) went wrong when a student encounters difficulties. Also, completing the entire problem solving in a single stage imposes a too-high cognitive load on students in their problem solving.

2. The systems have not been investigated empirically, and hence the effectiveness of applying the systems to actual teaching is unknown.

The purpose of this paper is to propose a new system (named MathCAL) that is based on the four problem-solving stages mentioned by Polya (1945): (1) understanding the problem, (2) making a plan, (3) executing the plan and (4) reviewing the solution. The system assists in achieving a successful outcome at each stage. For example, the schema representation and solution tree are used as assistants in stage 3.

The proposed system was evaluated by conducting an experiment using fifth-grade students in elementary school as subjects. The following two issues are explored here:

1. Comparing problem-solving ability before and after the experiment.
2. Whether assistance provided at various stages helped students with their problem solving.

2. Schema and solution tree

This system uses the schema representation to visualize concepts in the problems. Schemas are combined one by one to form a tree structure called a solution tree. This tree structure elucidates in detail the path taken to solve the problem, and thus provides students with a record of their problem-solving procedures.

2.1. Schema

A schema is a graphical representation consisting of two operand nodes; one operator node and one result node (see Fig. 1). Each operand and result node comes with two attributes, label and value, representing the meaning of the node and its numerical value, respectively. The values for the two operand nodes and the operator node correspond to the two operands and an operator in the mathematical expression. The value at the result node is the result of the expression. Therefore one schema may correspond to one expression, which in turn corresponds to one step in problem solving.
solving. The schema representation is very helpful for conceptualizing the semantics of the problem.

2.2. Solution tree

The solution tree is a tree structure comprising schemas that are interconnected via nodes having identical labels. This structure can not only describe the entire procedure of problem solving in detail, but can also help students execute a plan for solving the problem. Furthermore, students may use it to express their understanding of the problem. Fig. 2 is an example of a solution tree.

3. System outline

The functions in each stage are explained below.

3.1. Understanding the problem

In this stage, the system provides a “drawing pen” function for the student to highlight important information in the problem. For the instance of Fig. 3, the problem is displayed, and students can use the “select a pen” button to pick a pen for drawing lines on the important places in the problem.

3.2. Making a plan

The main function in this stage is providing the student with some built-in possible steps for solving the problem, from which the student selects the most appropriate one. As shown in Fig. 4, the screen contains four frames. The problem is displayed in the “problem frame” for the student to read. Built-in problem-solving steps prestored in the system are displayed in the “solving-step frame” as possible candidates for the student’s selection as he/she plans the order of problem-solving procedures. In Fig. 4, there are four built-in problem-solving steps in the
Fig. 2. An example of a solution tree.

(1) Number of peaches
(2) Price of each peach.
(3) How much do 34 peaches sell for?
(4) Number of rotten peaches.
(5) How much do 5 peaches sell for?
(6) How much money is made from the sales?

Fig. 3. The highlighting function.

(1) Mr. Chien had 34 peaches, and 5 of them were rotten. He sold all of the remaining peaches at $40 each. How much money did Mr. Chien get from the sale?
The "planning frame" is where the student plans his/her problem-solving steps and sequences. The student selects the built-in steps from the "solving-step frame", then drags and drops them into this frame according to his/her problem-solving order. In the example of Fig. 4, the student has chosen the first two steps in the "solving-step frame" as his/her solution plan.

The last frame is the "message frame" which displays feedback messages from the system. For example, after the student has pressed the "finish" button, the system compares the solution plan made by the student with the solution plan built into the system, and provides suggestions regarding the student’s problem solving and displays them in this frame.

3.3. Executing the plan

As shown in Fig. 5, the system provides three frames. Each of the problem-solving steps in the plan generated from the "making a plan" stage (see the "planning frame" in Fig. 4) is listed in the "planning frame" in the form of a "solution step" button. In the "planning frame" of Fig. 5, there are two "solution step" buttons. Whenever a student clicks a button, the system displays an empty schema in the execution frame, in which the student may enter related operands and an operator into the appropriate nodes. To fill in the operand, the cursor is moved over the position of the label attribute of the operand node in the schema. Right-clicking the mouse reveals a list of labels, one of which is selected and the student fills in a number for the value attribute. The operator is determined by using the "operator buttons" to input values into the operator node. After filling in the operands and operator, the student has to fill in the label attribute of the result node, and then
the value of the result is calculated automatically by the system and displayed in the result node. After finishing schemas corresponding to all “solution step” buttons, the system combines the schemas in the execution frame to form a solution tree as shown in the example in Fig. 6.

3.4. Reviewing the solution

In this stage the student fills in the expressions and answers as shown in Fig. 7. After filling in the blanks in the expression in the “evaluation frame”, the student presses the OK button which triggers the system to evaluate the results, and messages appear that indicate whether any mistake was made. Also, the student may press the “demo” button to see the correct problem-solving steps.

4. Experiment

4.1. Participants

The students participating in this experiment were 132 fifth-grade students selected from four classes in an elementary school in Taipei. The students, who were 11 years old in average, had recently learned basic calculation expressions involving the operations of addition, subtraction, multiplication and division. Students with lower problem-solving ability were determined on the following two criteria:
Fig. 6. Solution tree after combining schemas.

Fig. 7. Evaluation screen.
1. Those whose scores in the “mathematical problem-solving pretest” were lower than 10 out of a total of 15 points.
2. Those whose mid-term averages for Chinese and mathematics for the second semester of the fifth grade were lower than 60 points out of a total of 100.

We selected 49 students who met these criteria as participants in this study. Twenty-five of them were placed in the control group, practising mathematical problem solving using the written method, and the other 24 students were placed in the experimental group using the computer-assisted problem-solving system.

4.2. Experimental design

This study used a two-way mixed design. The between-group independent variable was group (or treatment), dichotomized into “not using the computer-assisted problem-solving system” (control group) and “using the computer-assisted problem-solving system” (experimental group). Participants are randomly assigned to two groups. The within-group independent variable is test, dichotomized into pre- and post-tests. The dependent variable was students’ scores in mathematical problem-solving pre- and post-tests.

4.3. Materials

4.3.1. Subject domain

This experiment used a fifth-grade mathematics textbook, from which we selected problems from six units, namely “the addition and subtraction of real fractions,” “the multiplication of fractions,” “pi, sectors and capacity,” “the area of triangles,” “unit cost and vertical planes” and “the area of a trapezium.” Related problems were collected, revised and compiled into a database of 80 problems as listed in Table 1 for practising with the MathCAL system.

4.3.2. Pretest and post-test

This experiment involved conducting tests to find out the changes in students’ mathematical problem-solving ability after practicing with the 80 problems. The questions were selected from the six units and organized into pretest and post-test, each of which contained 15 questions.

<table>
<thead>
<tr>
<th>Unit</th>
<th>Number of problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Addition and subtraction of real fractions</td>
<td>38</td>
</tr>
<tr>
<td>Multiplication of fractions</td>
<td>9</td>
</tr>
<tr>
<td>Pi, sectors and capacity</td>
<td>5</td>
</tr>
<tr>
<td>The area of triangles</td>
<td>5</td>
</tr>
<tr>
<td>Unit cost and vertical planes</td>
<td>21</td>
</tr>
<tr>
<td>The area of a trapezium</td>
<td>2</td>
</tr>
</tbody>
</table>
4.4. Procedures

The experiment spanned 6 weeks, with one pretest, one post-test and eight problem-solving practice sessions (twice a week, 40 min each).

One week before the formal experiment began, we gave the pretest to the students in the experimental and control groups (50 min) to test their mathematical problem-solving abilities, after which the formal experiment was performed. The students in the experimental group practiced using MathCAL, and the control group of students solved 10 mathematical problems on paper in each session. At week 6 the students in the experimental and control groups were given the 50-min post-test.

5. Results

5.1. Analysis of learning results

The means and standard deviations of students’ scores of the mathematical problem-solving tests are listed in Table 2. We conducted a two-way mixed design analysis of variance (ANOVA) on the pre- and post-test scores in the experimental and control groups. A significance level of 0.05 was adopted throughout the study. Bartlett’s homogeneity test was conducted and showed that the variances of the two groups were not heterogeneous \((p > 0.05)\), from which we concluded that the variances were homogeneous.

The results of ANOVA on the students’ mathematical problem-solving tests showed that the group factor was significant \((F(1,47) = 5.91, p < 0.05)\), indicating possible differences in the scores between the two groups. The test factor was significant \((F(1,47) = 4.22, p < 0.05)\), indicating that the pre- and post-test scores might be different. The interaction between groups and tests was also significant \((F(1,47) = 4.22, p < 0.05)\), which indicated that the magnitude of differences varied with level. The simple main-effect analysis showed that there was a statistically significant difference between the two groups in the mathematical problem-solving post-test \((F(1,94) = 7.54, p < 0.05)\) but not in the pre-test \((F(1,94) = 2.89, p > 0.05)\). We could therefore conclude that at the post-test, the problem-solving ability in the experimental group \((M = 9.18, SD = 2.72)\) was significantly better than that in the control group \((M = 7.02, SD = 2.80)\). The simple-main effect comparison between pre- and post-tests revealed a significant difference in the experimental group \((F(1,47) = 14.71, p < 0.05, M = 7.75, SD = 2.30\) and \(M = 9.18, SD = 2.72\) for the pre- and post-tests, respectively) but not in the control group \((F(1,47) = 0.96, p > 0.05, M = 6.66, SD = 2.40\) and \(M = 7.02, SD = 2.80\) for the pre- and post-tests, respectively), indicating that the training had resulted in significant progress only in the experimental group.

Table 2

Means \((M)\) and standard deviations \((SD)\) of the pre- and post-tests

<table>
<thead>
<tr>
<th>Group</th>
<th>(N)</th>
<th>Pretest (M)</th>
<th>Pretest SD</th>
<th>Post-test (M)</th>
<th>Post-test SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experimental</td>
<td>24</td>
<td>7.75</td>
<td>2.3</td>
<td>9.18</td>
<td>2.72</td>
</tr>
<tr>
<td>Control</td>
<td>25</td>
<td>6.66</td>
<td>2.4</td>
<td>7.02</td>
<td>2.8</td>
</tr>
</tbody>
</table>
5.2. Analysis of problem-solving stages

Log files on the problem-solving stages for all students were recorded by the system. The results were summarized in Table 3, from which we learn the following:

1. **Number of problems practiced.** On average the students in the experimental group completed 38.4 of the 80 problems compiled for this experiment. We discovered that the highest number of problems completed was 62, while the lowest was 8.

2. **Average number of steps for each problem.** In the “making a plan” stage the system provides some built-in problem-solving steps for each problem, from which the students can select the ones that may help in solving the problem. Table 3 shows that the number of steps used by the students for each problem (2.5) and the number of steps in the correct answer (2.2) were very similar.

3. **Highlighting.** Table 3 shows that the proportion of students who highlighted important information was quite low (16.7%). In 63% of cases this was the entire information, and in 37% of cases this was numbers or phrasal information. In other words, most cases involved highlighting in its entirety, which is no better than not highlighting. The highlighted content showed that some of the students used highlighting to help them identify the relevant information in the problems.

4. **Use of calculators.** Table 3 shows that only 8% of the students used calculators during their problem solving.

5. **Referring to correct answers.** Table 3 shows that 46% and 50% of the students referred to the correct answers in the “making a plan” and “executing the plan” stages, respectively. Although the students solved nearly half of the problems by referring to the system’s built-in answers, the significant difference between the problem-solving abilities in the experimental and control groups suggests that self-initiated learning took place while students were engaged in the process of referring to the answers.

6. **Constructing the solution tree.** Table 3 shows that only 42% of the students constructed complete solution trees, whereas we had expected that all of the students would review the problem-solving procedures for each problem through the solution trees after they had completed the “executing the plan” stage.

<table>
<thead>
<tr>
<th>Item</th>
<th>Mean or percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average number of problems worked on by students</td>
<td>38.4</td>
</tr>
<tr>
<td>Average number of steps taken by students for each problem</td>
<td>2.5</td>
</tr>
<tr>
<td>Average number of steps for correct answer</td>
<td>2.2</td>
</tr>
<tr>
<td>Planning time for each question</td>
<td>4.8</td>
</tr>
<tr>
<td>Highlighting (%)</td>
<td>16.7</td>
</tr>
<tr>
<td>Use of calculators (%)</td>
<td>8</td>
</tr>
<tr>
<td>Referring to the answer in the “making plans” stage (%)</td>
<td>46</td>
</tr>
<tr>
<td>Referring to the answer in the “executing plans” stage (%)</td>
<td>50</td>
</tr>
<tr>
<td>Constructing the solution tree (%)</td>
<td>42</td>
</tr>
</tbody>
</table>
6. Discussion and conclusions

Researchers have developed many types of problem-solving assistance based on difficulties discovered at various stages of the procedures adopted by students when problem solving. One of the most recommended problem-solving assistances is visualization. Although this study continues the convention of using such assistance, it included a few major differences from previous studies. The first is the application of different assistance at different stages of the problem-solving procedure. The second is empirically testing the effectiveness of a computer-assisted problem-solving system in improving students’ mathematical ability when it is applied to the classroom. We now discuss our conclusions from the experiment.

6.1. The effectiveness in improving students’ problem-solving ability

The effectiveness of many previous computer-assisted problem-solving systems, such as DISCOVER (Steele & Steele, 1999), has not been evaluated by actual use in the classroom. In our experiment the students in the experimental group showed not only significantly more improvement in the post-test than those in the control group, but also a significant difference in the pretest and post-test scores. In contrast, although the control group also showed improvement in test scores on the post-test, there was no significant difference between the pretest and post-test scores. The empirical results showed that even though the experimental group practiced with about half as many problems as the control group, the intervention of a computer-assisted problem-solving system improved students’ problem-solving ability.

6.2. The effects of providing assistance at the various stages of the problem-solving procedure

Cognitive load is one of the main factors considered when exploring teaching strategies or teaching material design (Paas, Van Merrienboer, & Adam, 1994; Sweller, Van Merrienboer, & Paas, 1998). A computer-assisted problem-solving system designed in stages can provide two advantages: (1) decreasing the cognitive load and frustration in learning through the system’s guidance and feedback and (2) improving students’ problem-solving skills by using a step-by-step approach.

In the studies by Derry and Hawkes (1993) and Reusser (1996) the entire problem-solving procedure, comprising understanding the problem, making a plan, executing the plan and reviewing the solution, was completed in just one stage, with no clear distinction between these steps. Although the DISCOVER system developed by Steele and Steele (1999) provides a teaching strategy to guide students in problem solving step by step, it has a text-based interface that does not utilize visualization. Moreover, since the system lists expressions and calculates answers directly, how a student’s problem-solving approach has changed in the stages of “making a plan” and “executing the plan” cannot be distinguished.

In summary, this study built on the results of previous studies and took them one step further. Integrating highlighting, visualized representation, solution review and other assistance in the problem-solving procedure, it developed a computer-assisted problem-solving system and tested it on elementary school mathematical problems that involve the operations of addition,
subtraction, multiplication and division. The system was empirically demonstrated to be effective in improving the performance of students with lower problem-solving ability.

Acknowledgement

This work was supported in part by the National Science Council and the Ministry of Education, Taiwan, ROC, under the contracts NSC 90-2520-S-003-006 and 89-H-FA07-1-4.

References


