Improving School Children's Mathematical Word Problem Solving Skills through Computer-Based Multiple Representations

Tufan Adiguzel
Texas A&M University

Yavuz Akpinar
Bogazici University

Abstract

Instructional resources that employ multiple representations have become commonplace in mathematics classrooms. This study will present computer software, LaborScale which was designed to improve seventh grade students' word problem-solving skills through computer-based multiple representations including graphic, symbolic, and audio representations. The proposed presentation will illustrate the design, implementation and validation of an interactive learning environment (ILE), LaborScale. This ILE is based upon the principles of computer based interactive problem solving environments, which connects different types of knowledge representation forms, and aims primarily to assist students as they explore symbolic representations used in word-problem solving process.

Problem Solving

When solving problems, a learner combines previously learned elements of knowledge, rules, techniques, skills, and concepts to provide a solution to a novel situation. It is generally accepted that mathematics is both process and product: both an organized body of knowledge and a creative activity in which the learner participates. It might, in fact, be claimed that the real purpose of learning rules, techniques, and content is to enable the learner to do mathematics, indeed to solve problems (Orton, 1987). Thus problem solving can be considered to be the real essence of mathematics. Gagnè (1985) has expressed the view that problem solving is the highest form of learning. Having solved a problem, one has learned. One might only have learned to solve that problem, but it is more likely that one has learned to solve a variety of similar problems and perhaps even a variety of problems possessing some similar characteristics. Jonassen, Howland, Moore, and Marra (2003) also point out that solving problems are meaningful kind of learning activity in educational settings.

Problem solving activities introduce difficulties for management by teachers. For example: choosing and sequencing problem solving tasks, determining the degree and type of assistance to be given to students, maintaining motivation, and knowing how to consolidate understanding through reflection and follow-up discussions, demand continuous decision making on the part of teachers. Similarly, learners have complicated tasks to complete during problem solving, especially when solving word problems. Word problems are set in specific contexts from which students have to develop representations. Learners have to link representations to appropriate mathematical formulations, and to apply appropriate techniques in order to produce a solution. Learners have to monitor and evaluate this problem solving process, and consider the implications of the solution. In consequence, giving assistance and managing problem solving in ways which create effective learning and the development of cognitive skills is not an easy task.

Research shows that a major source of difficulty experienced by children in the problem solving process is transforming the written word into mathematical operations and the symbolization of these operations. Namely, children are required to disembed the information from the problem context, select the relevant values, and insert them into some formula. However, children are not very successful in transferring their abilities to solve problems to subsequent problems (Jonassen et al., 2003). Orton (1987) indicated that the most common difficulty of problem solving is failure to use known information. He also noted that in order to cope with this difficulty pupils should (1) write the problem in primitive form and sketch an accurate picture of the setup (where applicable), (2) transform the primitive statements to simpler language, and (3) translate verbal problems to more abstract mathematical statement(s) and figures, diagrams, charts and other similar representations.

Multiple Representations

Recent approaches to mathematics instruction in the classroom emphasize mathematics as flexible, insightful problem solving that requires understanding that mathematics involves pattern seeking, experimentation, hypothesis testing, and active seeking of solutions. But children's beliefs about the nature of mathematics contrast with this emphasis. For example, Baroody (1987) asserts that due to an overemphasis on 'the right answer', children commonly believe that all problems must have a correct answer, that there is only one correct way to solve a problem and that inexact answers or procedures (such as estimates) are undesirable. In order to recognize that multiple solutions and different representations of problems are possible, children need to have higher order problem solving skills. Polya (1962) advocated that solvers should choose multiple representations when they begin to solve a problem. Jiang and Mcclintock (2000) also suggested that encouraging multiple solutions to problem solving plays an important role in facilitating students' understanding of mathematical concepts and their grasp of methods of mathematical thinking. In this way, the National Council of Teachers of Mathematics (NCTM, 2000) states, "representations should be treated as essential elements in supporting students' understanding of mathematical concepts and relationships; in communicating mathematical approaches, arguments, and understandings to one's self and to others; in recognizing connections among related mathematical concepts; and in applying mathematics to realistic problem situations through modeling" (p. 67).

Representations are mainly divided into two categories. External representations are the knowledge and structure in the environment, as physical symbols, objects, or dimensions and as external rules, constraints, or relations embedded in physical configurations (Zhang, 1997, p. 180). Internal representations are retrieved from memory by cognitive processes. External and internal representations are particularly beneficial for learning when they are multiple. In most cases, learners have to process multiple representations, including graphics, symbols and audio. Classroom teaching has traditionally employed multiple external representations (MERs) in the pursuit of helping students learn. Teachers use MERs explicitly in order to make abstract situations more concrete.

Kaput (1992) proposed that multiple linked representations might allow learners to perceive complex ideas in a new way and to apply them more effectively. By providing a rich source of representations of a domain, one can supply learners with opportunities to build references across these representations. Such knowledge can be used to expose underlying structure in the domain represented. According to this view, mathematics knowledge can be characterized as the ability to construct and map across different representations.

Computer-Based Multiple Representations

Computer environments have been gaining great importance in education. Numerical computation tools can be used by problem solvers to emphasize planning and interpretation of arithmetic operations. The existence of computer graphics tools can be used to help students understand abstract mathematical concepts, to create entirely new graphic oriented representations of traditional mathematical topics, or to provide alternative visual methods in mathematical problem solving. As Kaput (1992) states for the case of mathematics education, this entails that routine computations can be off-loaded to a machine, that new representational mechanisms only available on computers (such as programs as representations) become available, and that one can reify abstract concepts by means of computer simulations, making them more readily accessible for reflection and dialogue.

Fey (1989) asserts that the use of numerical, graphic and symbol manipulation is a powerful technique for mathematics teaching and learning. He identified several ways in which computer-based representations of mathematical ideas are unique and especially promising as instructional and problem solving. First, computer representations of mathematical ideas and procedures can be made dynamic in ways that no text or chalkboard diagram can. Second, the computer makes it possible to offer individual students an environment for work with representations that are flexible, but at the same time, constrained to give corrective feedback to each individual user whenever appropriate. Third, the electronic representation plays a role in helping move students from concrete thinking about an idea or procedure to an ultimately more powerful abstract symbolic form. Fourth, the versatility of computer graphics has made it possible to give entirely new kinds of representations for mathematics-representation that can be created by each computer user to suit particular purposes. Finally, the machine accuracy of computer generated numerical, graphic, and symbolic representations make those computer representations available as powerful new tools for actually solving problems (p. 255).

Designing the Interactive Learning

The use of multimedia technology has offered an alternative way of delivering instruction. The old text-based approach to learning is being superseded by an approach, which includes multisensory representations (Jonassen et al., 2003). Interactive multimedia is one of the most promising technologies of the time and has the potential to revolutionize the way we work, learn, and communicate (Macromedia, 1992; Staub & Wertherbe, 1989). Interactive multimedia programs take the idea of learning and doing seriously. With interactive multimedia programs, the learning process is modified by the actions of the learners, thus changing the roles of both the learner and the teacher. Interactive multimedia learning is also a process, rather than a technology, that places new learning potential into the hands of users (Jonassen, 1999). The ideal interactive learning environment (ILE), then, is one where students are encouraged to undertake such activities and are provided with feedback as they do so.

Brooks (1993) stated that, with all the additional capabilities of the growing number of multimedia applications, the design of these applications has become a nightmare. He also pointed out the preponderance of ugly interfaces containing screens full of multiple fonts, insignificant boxes, irrelevant noises, and confusing webs of possible interactivity among the features of poorly designed multimedia packages. There are many requirements that must be checked while designing an interface such as screen design, learner control and navigation, use of feedback, student interactivity, and video and audio elements (Stemler, 1997). So, the design of the interface, which considers interactivity, is clearly important (Frye et al., 1988). Hence a properly designed interface should make the cognitive process transparent and externalized so as to support evaluation, reflection, and discussion and direct accessibility.

The following principles should be considered during the design of ILEs (Akpinar & Hartley, 1996).

- The ILE should provide interactive objects and operators, which are visual and can be directly manipulated by pupils.
- The ILE system should provide mechanisms for pupils to check the validity of their methods, and thus receive some feedback on the appropriateness of their actions in relation to task.
- As the instruction aims to support links between the concrete and symbolic representation of word problems, the ILE should be able to display these forms so that the equivalence between is apparent. The system should also be able to move its presentation modes to the symbolic as students gain in competence.
- The ILE should allow experimentation of concepts and procedures in ways that relate to the children's experiences. In brief the ILE should be able to support guided discovery as well as directed methods of instruction.
- The ILE should allow the learning to be conceptualized and procedural in its approach, and be capable of adjusting to the task needs of teachers.

Interactive Learning Environment: LaborScale

The overall aim of this research was to investigate the design of the LaborScale ILE that can assist problem solving performance and understanding specifically mathematical work and pool problems. Problem solving requires the integration and utilization of multiple knowledge representations e.g. graphical, symbolic, and audio. Depending upon these factors, the design of the computer based learning environment must take into consideration students' knowledge so that it can accommodate different levels of competence and be useful for varying modes of instruction in the classroom (Mayer 1985). The proposed design to realize these aims is LaborScale that provides a constructive environment based on direct manipulation, user-system interactions and available interface design.

The ILE should have all the features and components to develop children's word problem solving skills. Hence, it should be designed with an object-oriented and direct-manipulation approach. In order to reach this objective, the instructional software (LaborScale) was developed and implemented by the researchers using an authoring tool, Asymetrix Toolbook II 5.0 and other related multimedia programs (Macromedia Flash 5.0, 3D Studio Max and Photoshop 5.0) to support Toolbook Application with videos, animations, audios and pictures. The user-friendliness of Toolbook interface and its accompanied object-oriented scripting language, Openscript, used to specify the functionality were well suited to the development and implementation of the prototype. Toolbook software is a development environment that provides tools to draw objects that can be made interactive using the Openscript programming language. Also the development can be carried out incrementally. Further, Toolbook is event-driven i.e. an application can respond to events such as mouse clicking whenever they occur. This is suitable for the ILE interface that is based on a direct-manipulation approach in addition to its ability to produce quality visual animation.

The user-interface of LaborScale has two-page design consisting of multiple viewers in which each viewer has a background and a foreground containing objects such as fields, buttons, graphics and text. The user interface has two units: a curriculum-manager unit and a student-working unit.

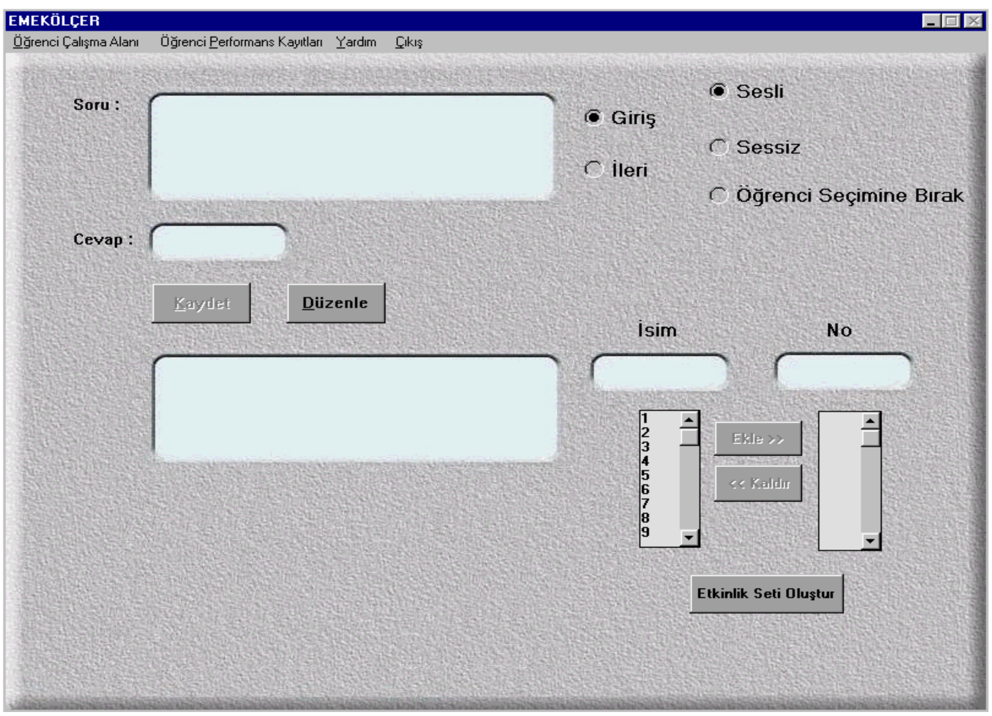
Curriculum Manager Unit

Curriculum Manager Unit (CMU) is one of the main windows of the LaborScale (Figure 1). This is the place where teachers set problems and customize environments for students. Each problem specification will need to provide context information, and the concepts based on the activity sets. Hence, the purpose of the problem specification is to provide contexts familiar to the students.

The CMU has been designed to manage the specification of activity sets that contains the problem content, specification of problems, and types of representation. In order to form a new activity set for any student, teachers can use two methods. One has two steps, which are pre-storing problems and their answers, and specification of the problems depending on the level of the students. The other is only specification of problems, which are saved to the system previously. In brief, teachers can manage the following tasks by using CMU:

- Forming problem sets including simple and advanced level problems of two types, work and pool, by saving problems to the systems,
- Setting audio environment of student-working unit,
- Preparing an activity set with respect to the students' level,
- Looking at the performances of the students who finish their activity sets,
- Viewing the activity set in the student-working unit.

Figure 1. A screen of curriculum manager unit

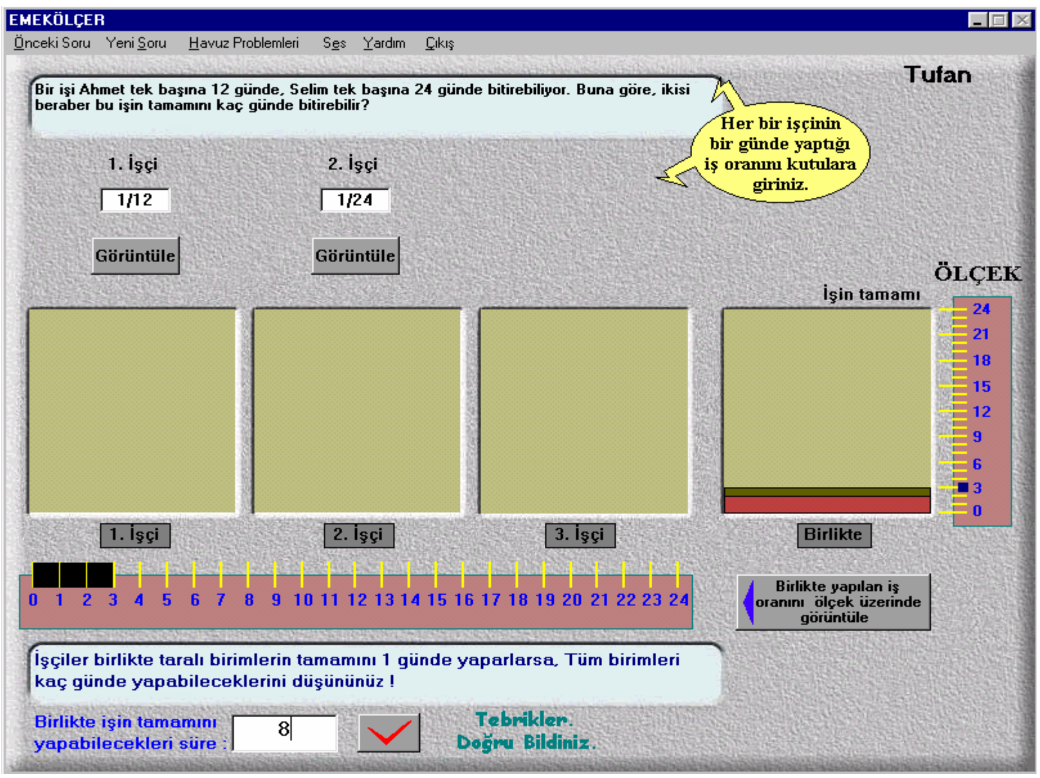


Student Working Unit

The student working window (Figure 2), the learner mode of the LaborScale, is the second main window of the LaborScale. The Curriculum-Manager Unit passes the sequence of problems to the ILE controller that is to manage the interactions with students and to keep records of their progress. Hence a principal consideration in the design of this unit was the user-system interface in which these interactions take place. LaborScale is based on a high degree of graphical and symbolic object manipulation, and with the interface users are able to directly manipulate the LaborScale objects, for example by giving the values symbolically, dragging and dropping of picture of these values, and combining the representations of them to reach a solution. To outline, students can manage the following tasks in this unit:

- Displaying ratios they entered in the problems,
- Dragging and dropping the displayed objects and displaying a vertical scale as a result of this,
- Reaching right answers of the problems by analyzing a horizontal scale depending on the vertical scale,
- Setting audio environment,
- Transition to other problems.

Figure 2. A screen of student working unit



Evaluation studies

The validation of LaborScale was carried out during last week of May and first week of June 2001. The method of the research was pretest and posttest group design. The sample of the study was selected from seventh grade students of two different schools (Public school, school A and private school, school B) that have a computer laboratory. The subjects were selected by using clustering sample technique, namely, one class was selected from each school. The validation experiment was performed as a pretest of the work and pool problems. The pre-test was administered to 80 students (59 from public school and 21 from private school).

Their average age was 14 and 40 of them were girls and 40 of them were boys. For the application and post-test, convenient students were chosen from each class by considering their teacher's opinions and number of computers in the computer laboratories of both schools. Students were chosen according to their achievements for each mode of the pre-test, namely, numerical solutions, symbolic and graphic representations of solutions. The students' actions in LaborScale were recorded by the system and notes were taken by the researcher as well as the pre and posttest differences in performance.

The instructor organized software for 27 students to run in their laboratories. The reason for choosing only 27 students was that there were 28 computers in the computer laboratory of School A, but 17 of them were available
to run the software. There were 13 computers in the computer laboratory of School B, but 10 of them were available to run the software. Five problems randomly chosen from nine problems were assigned to these students for both work and pool problems respectively by regarding the pre-test scores of the students. Pool problems were given to them at the second week of the application to allow students to manage problems at ease and meaningfully. For low-achievers, three of the problems were simple and two of them were advanced. For intermediate students and high achievers, two of the problems were simple and three of them were advanced.

Before the application of the software in both schools, the researcher provided students with an orientation session at which the students received explanations and were experienced on how they will use the program by presenting worked examples for solving different type of problems. This session lasted about 20 minutes. Since the students in both schools had a regular computer course, they had no difficulty in controlling and manipulating the environment during the instruction.

After an orientation session, all groups received computer-assisted treatment for two hours without any break for two weeks respectively. During the instruction, students were left alone and they only interacted with computers. They solved their own problems about the program by themselves except system problems by using the help and information modes of the software. Therefore, the researcher behaved like an observer in the application. The time that the students completed their activity sets changed between 30 and 65 minutes for work problems, however, the instruction on pool problems lasted between 20 and 45 minutes.

At the end of the instruction the performance test (PT) was conducted to all subjects as a post-test. After all, the questionnaire about the software evaluation was applied to the teachers involved in the study to obtain their criticisms.

In order to analyze the differences between the pre and post tests mean scores of the whole group and School A obtained from each mode of PT, paired sampled t-test was used, and Wilcoxon test was used for school B since the numbers of subjects in the groups were too small.

The pre and posttests results showed significant improvements in students' performances for each mode that pointed to the benefits of LaborScale ILE (See Table 1, Table 2, and Table 3). When the schools were analyzed separately, there was a significant increase in all modes of the tests for each school. Looking at the posttest results of each mode, significant improvements were also observed (for more details see Adiguzel, 2001).

Table 1 *Differences between pre-post test scores of numerical solutions to PT*

Groups	N	Pre-test Num. Sol. Mean	Pre-test Num. Sol. Std. Dev.	Post-test Num. Sol. Mean	Post-test Num. Sol. Std. Dev.	t	Sig. (2-tailed)
Total	27	22.85	27.28	85.78	21.70	10.997*	.000
School A	17	18.47	25.36	88.35	24.43	9.036*	.000
						z	
School B	10	30.30	30.16	81.40	16.29	2.805*	.005

*$p < .05$.

Conclusion

Studies formed a base for using computers featuring multiple linked representations to assist students with the transition from concrete experiences to abstract mathematical ideas, with the practice of skills, and with the process of problem solving, like in the LaborScale ILE, namely, beginning with the concrete representations and reaching the symbolic representations by using visual components supported by audio developed seventh grade students' performance on work and pool problems.

Table 2 *Differences between pre-post test scores of symbolic mode of PT*

Groups	N	Pre-test Symbolic Mean	Pre-test Symbolic Std. Dev.	Post-test Symbolic Mean	Post-test Symbolic Std. Dev.	t	Sig. (2-tailed)
Total	27	2.74	8.86	92.37	11.27	36.443*	.000
School A	17	2.47	9.68	96.59	5.49	37.343*	.000
						z	
School B	10	3.20	7.73	85.20	14.92	2.807*	.005

*$p < .05$.

Table 3 *Differences between pre-post test scores of graphic mode of PT*

Groups	N	Pre-test Graphic Mean	Pre-test Graphic Std. Dev.	Post-test Graphic Mean	Post-test Graphic Std. Dev.	t	Sig. (2-tailed)
Total	27	18.67	18.62	82.26	31.25	10.380*	.000
School A	17	21.65	14.71	95.00	10.90	16.858*	.000
						z	
School B	10	13.60	23.92	60.60	42.21	2.499*	.012

*$p < .05$.

In the post-test, students had different graphical representations of solutions. The results showed that they were affected from the visual components of the LaborScale ILE, namely, some of them drew a box, meant whole work, for each worker and they indicated the results by the graph corresponding these related graphs. Some of them drew a vertical scale whose pointer's location showed the values of work done in a day for each worker and the result. The rest of the students drew a horizontal scale, similar to the number line, whose pointer's location also represented the values of work done in a day for each worker and the result. This proves that if the students are given more visual representations, they will use and connect them to the symbolic representation of these and they will grasp the meaning of the word problem solving by concretizing them.

The increase in the symbolic mode was more than the other modes. This proves that multiple linked representations relating the symbolic representation to graphic representation allow learners to perceive complex ideas in a new way and to apply them more effectively. However, the increase in the graphical mode was significant but less than the other modes. The reason for this may be that since text-based books were dominant in the curriculum, children have not developed to present problems graphically. However, the graphical representations need to be well constructed and be capable of representing the information in a problem to enable the processing capabilities of the human visual system to be exploited, so that perceptual features and judgments can be developed and related to a more abstract symbolic understanding (Cox & Brna, 1995). Also, graphical representations are effective problem solving and learning tools because they reduce the space of applicable operations and they are more specific than the other representations. Relating with the theory, good performance on finding and presenting a graphic representation of a solution raised the performance on grasping symbolic representation of the solution in the study.

A similar result to the findings of this research was the outcome of ANIMATE software (Nathan, 1991). However, the potentials and facilities of ANIMATE differ from the LaborScale. Though Nathan's ANIMATE can not generalize the solution method into an algebraic formula, LaborScale can help students to build up algebraic formula that may be generalized and employed in a wide variety of problems
During the application, the time at which students finished each activity set was recorded by the researcher. According to the results, students spent longer time period on the first question of the activity set of the work problems than the other questions of the activity set since they were adapted to the system in the first question. After they gained an experience on the work problems, they acted carefully and swiftly on the pool problems

since the application of the pool problems was applied to them one week later.

Performance recording unit of the program stored two main functions of the students: three trials on the values that are work done in a day for each worker and the answer of the problem for each trial. According to the records obtained from this unit, all students had tried at least three times on the advanced level problems and problems requiring complex calculation to reach a right answer. The reason for this was the greatness of the least common multiple related with the value of denominator of the work done in one day and that because of the big unit differences in the scale students could not grasp net measurement. However, they usually solved these problems after they tried at least three times. The main thing in this part is requiring the students to grasp the values from the problem. As a result, students' problem solving skills significantly improved by the help of instructional software, LaborScale.

The rationale of the development of the LaborScale was to base students' problem solving on multiple representations. This should aid understanding, conform to problem solving as an investigatory and creative activity. While LaborScale was successful in fulfilling many of these claims in its design and conception, there are further requirements to complement or supplement its current facilities and strengths.

In conclusion, the LaborScale software and validation studies have given some support to the design principles of ILEs in which the aim is to produce user-system interfaces that release students' knowledge and stimulate active and investigatory methods of learning. The suggested further work could re-illuminate design principles for multi-representational interfaces. And, since the research was concluded successfully and significant results were obtained, hopefully it should be adapted for other domains of mathematics.

References

Adiguzel, T. (2001). Developing school children's word problem solving skills through computer based multiple representations. Unpublished master's thesis, Bogazici University, Turkey.

Akpinar, Y., & Hartley, J. R. (1996). Designing interactive learning environments. Journal of Computer Assisted Learning, 12(1), 33-46.

Baroody, A.J. (1987). Children's mathematical thinking: A developmental framework for preschool, primary, and special education teachers. New York: Teachers College Press.

Brooks, R. M. (1993). Principles for effective hypermedia design. Technical Communication, 40(3), 422-428.

Cox, R., & Brna, P. (1995). Supporting the use of external representations in problem solving: The need for flexible learning environments. Journal of Artificial Intelligence in Education, 6(2), 239-302.

Fey, J. T. (1989). Technology and mathematics education: A survey of recent developments and important problems. Educational Studies in Mathematics, 20, 237-272.

Frye D., Littman D., & Soloway E. (1988). The next wave of problems in ITS: Confronting the "user issues" of interface design and system evaluation. In J. Psotka, D. L. Massey, and S. A. Mutter (Eds.) *Intelligent tutoring systems: Lessons learned*. Lawrence Erlbaum. Hillsdale, NJ.

Gagne, R. M. (1985). *The conditions of learning and theory of instruction*. New York: CBS College Publishing.

Jiang, Z., & McClintock, E. (2000). Multiple approaches to problem solving and the use of technology. *Journal of Computers in Mathematics and Science Teaching, 19*(1), 7-20.

Jonassen, D. H. (1999). Designing constructivist learning environments. In C. M. Reigeluth (Ed.), *Instructional-design theories and models: A new paradigm of instructional theory* (pp. 215-239). Mahwah, NJ: Lawrence Erlbaum Associates.

Jonassen, D. H., Howland, J., Moore, J., & Marra, R. M. (2003). *Learning to solve problems with technology: A constructivist perspective*. Pearson Education Inc.

Kaput, J. (1992). Technology and mathematics education. In D. A. Grouws (Ed.), *Handbook of research on mathematics teaching and learning* (pp. 515-556). New York: Macmillan.

Macromedia, Inc. (1992). *Action! User guide*. San Francisco, CA.

Mayer, R. (1985). Implications of cognitive psychology for instruction in mathematical problem solving. In E. A. Silver (Ed.), *Teaching and learning mathematical problem solving: Mulitiple research perspectives* (pp. 123-138). Hillsdale, NJ: Lawrence Erlbaum Associates.

Nathan, J. M. (1991). *A theory of word algebra problem comprehension and its implications for the design of computer-based learning environment*. Unpublished Ph.D. Dissertation, University of Colorado.

National Council of Teacher of Mathematics (2000). *Principles and standards for school mathematics.* Reston, VA: NCTM.

Orton, A. (1987). *Learning mathematics: Issues, theory and classroom practice*. London: Cassell.

Polya, G. (1962). *Mathematical discovery.* New York: John Wiley & Sons.

Staub, D.W., & Wetherbe, J.C. (1989). Information technologies for the 1990s: An organizational impact

perspective. *Communications of the ACM, 32*(11), 1328-1339.
Stemler, L. K. (1997). Educational characteristics of multimedia: A literature review. *Journal of Educational Multimedia and Hypermedia, 6*(3/4), 339-359.
Zhang, J. (1997). The nature of external representations in problem solving. *Cognitive Science, 21*(2), 179-217.