JAMES T. FEY

TECHNOLOGY AND MATHEMATICS EDUCATION: A SURVEY OF RECENT DEVELOPMENTS AND IMPORTANT PROBLEMS

ABSTRACT. This paper, prepared for the survey lecture of theme group T2 at the Sixth International Congress on Mathematical Education in Budapest, gives an overview and analysis of recent progress in applying electronic information technology to creation of new environments for intellectual work in mathematics. The paper is divided into six major sections considering the impact of: 1. Numerical computation; 2. Graphic computation; 3. Symbolic computation; 4. Multiple Representations of Information; 5. Programming and Connections of Computer Science and Mathematics Curricula; 6. Artificial Intelligence and Machine Tutors.

One of the most important tasks in mathematics education today is the revision of curricula and teaching methods to take advantage of electronic information technology. Developments in this decade alone have presented us with inexpensive and powerful hardware and software tools that challenge every traditional assumption about *what we should teach, how we should teach,* and *what students can learn.* This paper, prepared for the survey lecture of theme group T2 at the Sixth International Congress on Mathematical Education in Budapest, gives an overview and analysis of recent progress toward realization of the stunning promise in these new environments for intellectual work.

There is no shortage of speculative writing on the promise of revolution in school mathematics following from application of various calculating and computing tools to teaching, learning, and problem solving. Since ICME V in Adelaide, there has been a profusion of conference reports and position papers outlining potential technology-based innovations (for example, *School Mathematics, New Ideas with Computers,* 1987). An exciting array of experimental projects have begun demonstrating the prospects for mathematics classrooms that use calculators, computers, and videodisks to change the goals of curricula and longstanding patterns of teacher/student interaction.

The reports of ICME V itself (Mohyla, 1985) include hints of nearly every proposal and project reported since 1984. However, it is very difficult to determine the real impact of those ideas and development projects in the daily life of mathematics classrooms, and there is very little solid research evidence validating the nearly boundless optimism of technophiles in our field. In preparing this survey paper, I scanned a broad sample of published

Educational Studies in Mathematics **20**: 237–272, 1989.

work in the theme area, consulted with knowledgeable individuals from all over the world, and sifted through the proposals and papers from speakers in our theme group sessions at the Budapest Congress. It is clear that in every country and region there are many mathematics educators actively exploring technology prospects. However, it is also clear that a comprehensive and up-to-date description of such a vast and fluid field is really impossible. Much of the most exciting work is in progress, not yet in published journal articles or books, and the hardware/software environments available for such experiments are constantly improving. In the face of these conditions I have chosen not to attempt a true international survey, but to discuss the major problem/opportunity areas and to give illustrations of some of the most informative or thought-provoking work in each.

There are several possible ways to impose order on the array of technology-motivated ideas in mathematics education today. One is to look at the major tasks involved in school mathematics – selection of content and process goals, organization of teaching/learning environments, and assessment of achievement – and to describe the impact of technology on each. There are many suggestions and active development projects working on each of these dimensions of the problem.

1. Content/Process Goals – The most prominent technology-motivated suggestions for change in content/process goals focus on decreasing attention to those aspects of mathematical work that are readily done by machines and increasing emphasis on the conceptual thinking and planning required in any tool environment (Bjork and Brolin [4]; Brolin and Greger, 1987; Corbitt, 1985; Cornu [8]; Fey, 1984).

   Another family of content/process recommendations focus on ways to enhance and extend the current curriculum to mathematical ideas and applications of greater complexity than those accessible to most students via traditional methods (Fey, 1989; Roberts and Barclay, in press; Tinker [22]).
2. Teaching/Learning Styles – Many mathematics educators have looked at the new information processing tools and envisioned striking change in traditional teaching-learning patterns of mathematics classes. They see teachers shifting their roles from expositor and drill-master to tasksetter, counselor, information resource, manager, explainer, and fellow student, while students engage in considerably more self-directed exploratory learning activity (Fraser, 1986; Fraser *et al.* [13]; Klep [16]; Schoenfeld, 1988).

> The most common strategy for creating these new kinds of teaching/learning interactions is provision of some sort of computer microworld. In such a microworld mathematical or real world objects, relations, and operations are represented electronically in ways that permit controlled exploratory manipulation and observation of properties by learners searching to abstract underlying mathematical principles.

Unfortunately, while there is a certain naive logic to sorting ideas by focus on content or pedagogy, very few proposals or projects can be easily categorized in this way. Nearly every development program has an agenda of goals that imply changes in both content and pedagogy of school and university mathematics. To focus attention on **technology** prospects and the implications of those prospects for school mathematics, I have chosen to approach the survey task by looking at the different things that calculators and computers can do and to analyze the implications of each for both content and pedagogy in mathematics. Thus the paper is divided into six major sections considering the impact of:

1. Numerical computation
2. Graphic computation
3. Symbolic computation
4. Multiple Representations of Information
5. Programming and Connections of Computer Science and Mathematics Curricula
6. Artificial Intelligence and Machine Tutors

Each section describes some typical or particularly interesting studies. But it is important to keep in mind that those examples are at best only a small sample of what is going on in this exciting arena today.

NUMERICAL COMPUTATION AND MATHEMATICS EDUCATION

Development of modern digital computers was stimulated by important problems that could be solved effectively only with methods that involved extensive numerical/logical calculation. These numerical methods first influenced mathematics education as enhancements or alternative approaches in university courses like numerical analysis or linear algebra. But miniaturization of the computing technology led to a variety of personal numerical calculating tools that are now routinely available at low cost. This availability of handheld scientific calculators has forced reconsideration of curricular objectives in every topic that involves numerical computation.

Calculators and Arithmetic

The influence of calculators on objectives and teaching in arithmetic has been debated for nearly two decades. Proponents of calculator usage have seen great potential for shifting curricular emphasis from computational procedures to problem solving and to the mental arithmetic needed for estimation and checks on calculator results. However, skeptics have worried that people will become dangerously dependent on technology and that they will lose skills and understandings that play important roles in more advanced mathematics. As a matter of practical school curriculum policy, this debate is far from over. In most elementary schools around the world, use of calculators for instruction and testing is not routinely permitted. However, a growing body of research suggests that, when used wisely, calculators can enhance student conceptual understanding, problem solving, and attitudes toward mathematics – without apparent harm to acquisition of traditional skills. Those are the conclusions reached in a meta-analysis of 79 studies by Hembree and Dessart (1986), in a series of studies reported by Brolin (1987), and in a 3-year study of several thousand students reported by Wynands (1984). In fact, for many researchers the question of whether to allow calculator use seems to be settled in the affirmative and interest in such broad calculator studies has diminished. More typical of current research are studies that search for effective new ways to use calculators for instruction. For instance, Meissner (1987) has investigated the effects of calculator use on spontaneous development of student problem solving strategies.

The many studies on effects of calculator use give confidence that the mere presence of calculators in school mathematics will not inevitably lead to damaging consequences. However, in almost all of those experiments the calculator was used to complement instruction in traditional arithmetic skills – including memory of basic arithmetic facts and performance of traditional paper-and-pencil algorithms for computation. What remains an open and very important problem is determining the consequences of more daring experiments in which students are taught to rely more heavily on calculator help with arithmetic calculation or even basic facts of addition, subtraction, multiplication and division. Some major development projects have begun construction of elementary curricula that have goals which are appropriate for students who will live in a world where calculators are nearly always available at negligible cost, but it is too early to tell whether those projects will really push the limits of reducing traditional computation skill goals or what the effects of such changes might be.

Numerical Approaches to Algebra and Analysis

The numerical assistance provided by calculators offers a clear and attractive alternative to traditional paper-and-pencil skills wherever arithmetic computation is required – in elementary arithmetic, algebra, geometry, trigonometry, statistics, or calculus. What is less well appreciated is the role that ease of numerical computation can play in development of conceptual understanding in those advanced topics. The instructional power of numerical approaches to advanced mathematics is the theme in a series of development projects led by Leitzel, Osborne, Demana, Damarin, and Waits. They have focused attention on using a variety of numerical computation tools in helping students make the transition from arithmetic to algebraic reasoning. To establish solid intuitive understanding of variables and functions, students work with calculators and a simple spreadsheet computer program to investigate relations among variables. The basic strategy is to emphasize the search for patterns in tables of values for related numerical variables as a first step toward formal algebraic expression of such relationships. Research results suggest that this computation-rich *transition* to the abstractions of algebra is strikingly more effective than traditional approaches (Demana and Leitzel, 1988). Similar numerically-intensive approaches to calculus are also proving effective.

Quantitative Complexity

Application of computational tools to more advanced concepts of mathematics shows its clearest advantage in any situation that involves many interrelated variables or large sets of data. For instance, a variety of approaches have been developed for application of computing to work in *linear algebra* (Orzech, 1988). With tools available to perform basic matrix operations, it is quite reasonable to shift the focus of instruction from training in execution of the various algorithms to planning and interpretation of those operations. The vector-oriented language APL has gotten special attention in this area, but there are now several higher-level utilities, like Matlab or muMath, available on microcomputers. Some people have also explored use of spreadsheets to organize and execute operations like Gaussian elimination. Standard spreadsheet row and column operations correspond easily to the needed matrix procedures. The value of using these matrix tools, in either "black box" or "user guided" models, is supported mostly by anecdotal evidence, and it seems clear that the problem of finding an appropriate balance between skills that must be acquired by each student and skills for which students can rely on a machine is a crucial question, parallel to the questions about calculators and arithmetic.

The second major area in which computing promises help dealing with computationally complex situations is *data analysis* and *statistics*. Access to computers allows students to work with interesting and realistic collections of numerical data. The statistics education community has been very active in exploiting this opportunity. There are computer networks giving users access to interesting data sets for instruction, there are many useful pieces of software available for basic data analysis procedures, and curriculum development projects are showing how to blend these information processing resources into instruction at all grade levels (Friel and Russell [14]; Scheaffer, 1987; Swets, Rubin, and Feurzeig, 1987). Perhaps because data analysis and statistics are not viewed as traditional skill-based strands in the curriculum, the debate over potential risks of computer use seems not to have been as sharp as in other topics. The appeal and promise of expansion in this general area seems immense.

Discrete Numerical Methods

In one sense the use of numerical methods in traditional areas of mathematics, like calculus, can be viewed as a process of making estimates on the way to or in lieu of exact results. However, there is now growing interest in reformulating problems traditionally treated by continuous variables and calculus in the language of discrete processes – principally difference equations. Such approaches offer hope of illuminating or avoiding the difficult concepts related to limits in continuous mathematics. They lead naturally to the computer solution methods that are used so often in practical work, and they reinforce important concepts of computer science like recursion. While it seems unlikely that the traditional approaches of analysis will be rejected soon in favor of computer-based finite difference methods, there is a vigorous interest in exploring the alternatives and there are likely to be interesting development projects in the years just ahead (Sandefur and Vogt, 1988; Winkelmann, 1984).

Summary

Calculator and computer aided numerical methods are standard tools of mathematical work at all levels. However, when it comes to decisions about their role in curriculum and teaching at the school level, there remains considerable controversy about their potential impact. Despite a variety of research and development projects showing that access to numerical computation tools enhances learning and extends problem solving power of most students, without obvious deleterious effects, tradition dictates that use of those tools is permitted only *after* students have acquired some measure of personal skill in the procedures that have been automated in the machines.

Computer numerical tools like spreadsheets, vector and matrix operators, and statistical data analysis utilities provide attractive opportunities to enrich teaching of concepts and to extend the reach of problem solving in secondary school and university mathematics topics. But, with the exception of statistics, these tools seem to have made little significant impact on standard curricula. In fact, the use of computers for numerical investigations of mathematical concepts seems recently to have been overshadowed by the attraction of dynamic color graphic displays representing the same ideas. However, there is a growing body of research showing that neglect of numerical investigations deprives students of an important perspective on many mathematical ideas.

GRAPHIC COMPUTATION AND MATHEMATICS EDUCATION

The first uses of computers relied on their role as aids in numerical calculation, but the most appealing development in the past decade has been the use of computers as tools for creating and manipulating graphic images. When microcomputers made it possible to draw geometric shapes in two and three dimensions, to graph functions defined by algebraic rules, and to display diagrams consisting of user-defined icons, a whole new set of potential computerists were attracted to the machine. Students were fascinated by the dynamic graphic images of video games, and teachers saw the potential to give visual representations for abstract mathematical ideas. That promise is now being realized in a number of exciting development projects, but we are also discovering important limits to the initial enthusiasm.

Drawing Tools for Geometry

At the time of ICME V in Adelaide, there was tremendous enthusiasm for computer graphics in geometry – most centered on application of the turtle graphics mode in Logo. Turtle geometry continues to be an attractive vehicle for informal classroom development projects intended to enhance students' general reasoning abilities. However, there is also an emerging trend in Logo research toward studies that make more focused application of Logo characteristics to teaching of specific mathematical concepts, principles, and reasoning abilities.

In studies typical of this new trend: (1) Campbell (1987) found that even for very young children Logo explorations yielded significant payoffs in growth of student ability to estimate length; (2) Noss (1987) found that Logo experiences had significant effects in development of student intuitive understanding about angles; and (3) Clements (1987) found a variety of positive effects from Logo programming experiences. At the sessions of ICME VI a variety of short presentations indicated other ways that the Turtle geometry mode of Logo can be used to good effect in helping students discover important principles of plane geometry.

Extensions of turtle drawing to three dimensions have been developed by several investigators, with the hope that such dynamic explorations would have good effects on student abilities to interpret and construct planar representations of objects and motions in space. For instance, Cesar [7] is using a 3D-Turtle called the SEA-TURTLE to help students develop their perception and drawing of three dimensional figures. The SEA-TURTLE commands allow students to draw pieces of a solid figure in the plane and then to assemble them in space using commands like PITCH, BEND, ROLLRIGHT, and ROLLLEFT. This geometry tool is being used in teaching experiments which are assessed with measures of spatial intuition.

The turtle geometry feature of Logo has been recommended for use at a variety of grade levels. It has been used mainly in primary and early secondary years as a microworld for exploration and discovery activities. However, while teachers praise the classroom environment and imaginative student products of those Logo experiences, turtle geometry seems yet to have gained widespread acceptance as a standard feature of the mathematics curriculum in elementary or secondary schools. The dynamic differential geometry point of view that is at the heart of turtle programming has gained very little headway in shaping approaches to school geometry. In fact, the geometry microworlds that have gotten most attention at the secondary level are essentially electronic representations of the geometric constructions and measurements that students have done with paper-and-pencil for centuries.

One of the best-known of these new geometry microworlds is the Geometric Supposer series (Yerushalmy and Houde, 1986; Schwartz 1987). The Supposer programs help students make and test conjectures about properties of basic geometric figures by simulating the sorts of drawings and measurements that a mathematician would make in the course of a search for patterns. For instance, in a typical session the student might ask the computer to draw a triangle, to connect the midpoints of two sides, to measure the sides of the original triangle and the segments into which the midline divides two of those sides, to construct various ratios of the side measurements, and then to measure the various angles in the figure. To test the generality of any patterns noted, the user could ask the computer to repeat the sequence of procedures on several other triangles. This tool and the environment it creates have two principal objectives: To restore the act of discovery and conjecturing to a geometry course that has become a deadly routine of proving things that have been well known for centuries, and to facilitate inductive reasoning by making multiple tests of conjectures easy to execute. Reports from users of the Supposer are enthusiastic about its effects on classroom activity and on student beliefs about the nature of mathematics. Yerushalmy and Houde (1986, pp. 421–422) found that

> The pedagogy we used most closely resembles the teaching ordinarily found in science classes, where the primary focus is on the scientific process of collecting data, conjecturing, and finding counterexamples or generalizations. . . . Students spent the majority of class time discussing and doing geometry rather than listening to a teacher talk about it.

While it might be suggested that the operations described in Supposer sessions are really no more than what a careful student could do with paper-and-pencil and drawing tools, the fundamental difference is allocation of student time. With this kind of computer assistance, the student spends nearly every minute of a session planning or reflecting on results of some construction. With no help in the drawing and measuring, nearly every minute would be consumed by those tasks alone.

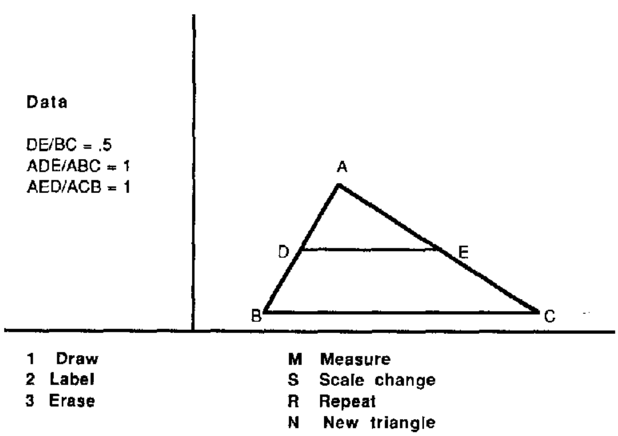


Fig. 1. A typical Geometric Supposer screen display.

The Supposer family of programs were written for use on the Apple II series of computers which, while impressive for their time, have graphics limitations that constrain the tool well short of what geometry teachers could easily envision and hope for. In France, the Cabri project has produced a similar but apparently more flexible geometric exploration tool for use with Macintosh computers. Cabri-geometry permits construction, transformation, and exploratory manipulation of most figures encountered in secondary mathematics. The central idea of the Cabri tool is that the user should be able at any time to modify the characteristics of any element of a figure and to see the resulting redrawn figure immediately. As with the Supposer, this encourages examination of many variations on a single theme with the goal of discovering invariant properties of all (Bellemain [3]).

In another similar geometry microworld called GeoDraw (Bell, 1987), the student has similar help with traditional constructions and measurements, but the additional option of doing Turtle-style vector drawing, plane transformations (available, but in a very limited way on the Supposer), and coordinate-based studies as well. A fourth example of geometric tool software (Kramer, Hadas, and Hershkowitz, 1986) presents students with a menu of basic construction options from which they are to assemble the solutions to give construction problems.

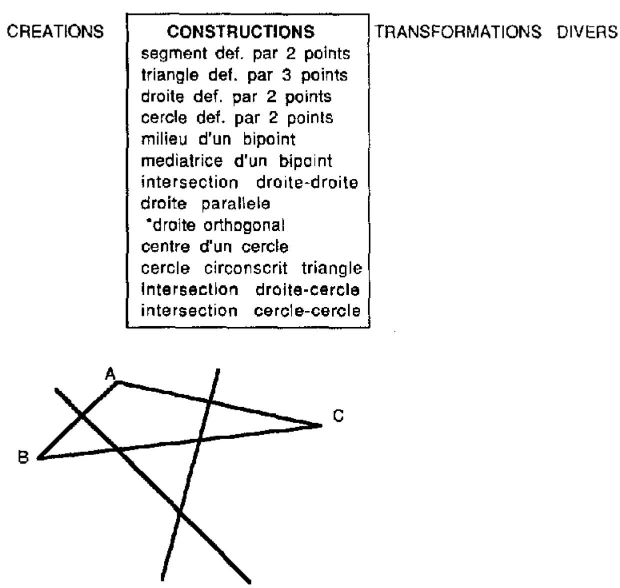


Fig. 2. A typical Cabri-Geometre screen display.

In reflecting on the potential impact of geometric software tools like these, Schwartz suggests that "the larger idea that underlies this program, i.e., that students can make their own mathematics and the microcomputers can help them to do so, can change the way mathematics is taught and learned at all levels" (1987, p. 635). Many who have studied and used the tools believe that he is right, but, like many other computer-motivated changes, implementation will require change in long-standing habits of teachers and in the expectations we have for outcomes of familiar courses of study.

Throughout the discussion of computers and geometry, there tends to be a tacit acceptance of traditional outcomes for geometry instruction. The computer is seen as a potentially powerful aid for teaching and student exploration, but the question is seldom turned the other way around. Is it possibly the case that in order to provide students with the conceptual tools they need to interpret and construct computer graphic images in problem solving we will have to focus on different principles and methods in our mathematics courses? What about fractals or the variety of discrete graphs/networks and other schematic diagrams that are now used as visual models of important systems, but are hardly decomposable into familiar euclidean figures and relations (Gaulin and Puchalska, 1987)? This seems to me a significant question that has not yet been addressed in any detail.

Graphic Images of Quantitative Relations

One of the most productive methods of mathematics is the representation, in coordinate graphs, of relations among quantitative variables. That is certainly one of the first applications that mathematicians try when they begin work with computers, and we all believe that such pictures of algebra ought to have very powerful effects on student understanding. Suggestions of ways to use such graphs have appeared in many places and the software tools available to facilitate graphing are really quite versatile and easy to use.

Typical "function grapher" software allows the user to enter rules for one or more functions or relations, to choose domain and range for graphing, and then to watch as high resolution color graphs are plotted. To focus attention on a smaller or larger window into the graph, various changes of scale are easy to command. In some experimental software the user can vary parameters in the function rule and observe resulting transformations of the graph or, conversely, transform the graph by dragging it with a mouse and observe simultaneous changes in the function

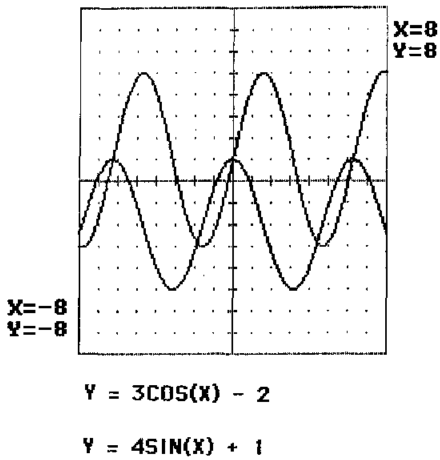


Fig. 3. A typical screen display for computer function graphers.

rule. In addition to the basic function graph itself, many programs will give a simultaneous plot of the derivative or integral. Software for three dimensional graphs of functions is not yet as well developed for microcomputers, but as standard machine memories and processing speeds increase, this tool will become standard as well. In fact, the conventional two-dimensional graphing is already available on several handheld devices [Casio fx-7000 and HP-28S], offering the hope that such machine capability can be available to students at all times for mathematical work.

In addition to the conventional graphs of functions with formal rules, science educators have devised laboratory interface hardware and software that permits real-time graphing of data from scientific experiments. Readings from temperature probes or light transducers can be translated directly into graphs of temperature, distance, velocity, or acceleration over time while users watch the experiment and the unfolding graph simultaneously (Tinker [22]). While the "real-time" aspect of this graphing software is especially attractive, it is, of course, possible to simulate such events with suitable programs. The Shell Centre's Eureka and Bottles programs, are among the best known of this software type (Fraser, 1986).

In more advanced topics, computer graphics have been particularly popular enhancements to the study of differential equations. Several teams have produced software that displays direction fields for given differential equations and then traces particular solution graphs from given initial conditions (Danby, 1988; Artigue, Gautheron, and Sertenac [1]).

Despite all the promise in these new dynamic graphic tools, it is reasonable to ask how they are being used to change mathematics teaching and learning. First, nearly everyone hopes that ready access to such graphs will enrich student understanding of algebraic forms – giving visual images of symbolic information. Initial instructional applications of function graphers usually involve simply displaying series of graphs for related function rules to reveal the patterns associated with various rule types and the effects of parameters in each type. Next students are asked to construct rules for functions that fit given graphic patterns. The Green Globs program is among the best known and highly praised examples of this software (Dugdale, 1984, 1987). In the Globs game, students must use their knowledge of algebraic forms to construct rules for graphs that hit "globs" on the coordinate screen. Variations on this theme present a systematic approach to graphing via transformations of basic forms, using computer displays to demonstrate the effects of the various transformations (Bloom, Comber, and Cross, 1986), and Tall (1985) has developed a highly regarded graphic-oriented approach to calculus. All such graphing activities are reported to be popular and successful, but hard research data supporting the claims is very sparse.

Much use of computer graphics has focused on using the computer to enhance understanding and skill in traditional mathematical topics. However, there is a line of work exploiting the new graphic tools that proposes significant changes in emphasis and goals also. In conventional pre-computer mathematics curricula, a great deal of time is devoted to theory and technique for construction of graphs for algebraic expressions of various types. There is an implicit assumption that once the graphs are produced it is a simple matter to use the information they represent to

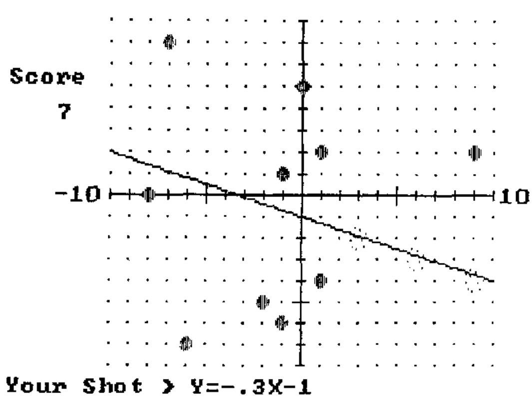


Fig. 4. A typical Green Globs screen display.

answer a variety of questions equivalent to solving equations or inequalities, determining maxima or minima, and studying rates of change. Of course, now it is not really essential for students to be so clever in constructing graphs by hand. Furthermore, a series of probing investigations have revealed that many students have a great deal of trouble interpreting the graphs that mathematicians use effectively, almost without thinking. This situation has led many mathematics educators to urge that the focus of graphing activity be shifted from construction to interpretation of graphs. Experiments with this theme have produced a number of very interesting effects.

First, when computer-generated graphs become the focus of classroom discussion about mathematics there is a notable change in the roles and interactions of teachers and students. For many students, traditional mathematics is perceived to be a formal game played according to arbitrary rules – a contest between teacher and student in which the challenge is to figure out secrets that the teacher keeps hidden. When students see a live experiment produce a computer graph (and even manipulate the experimental conditions themselves) or when students see a function rule typed into the computer and the related graph emerge immediately, they are presented with powerful illustrations of mathematics as a source of models for real phenomena. Furthermore, the classroom becomes a setting for student and teacher collaboration in the attempt to make sense out of the mathematics that is displayed before them.

The teacher role shifts from demonstration of "how to" produce a graph to explanations and questions of "what the graph is saying" about an algebraic expression or a situation it represents. Student tasks shift from plotting of points and drawing curves to writing explanations of key graph points or global features. In much the same way that numerical computation tools give an opportunity to emphasize planning and interpretation of arithmetic operations for problem solving, the existence of computer graphic tools can be used to revise the balance between conceptual and procedural knowledge in mathematics or to create entirely new graphic-oriented presentations of traditional mathematical topics. Examples of the first such application are given by a variety of computer-oriented curriculum development projects. The second type of graphics-oriented curriculum change is illustrated by the work in West Germany leading to a course called "Elementary Analysis" in which the basic concepts of limits, differentiation, and integration are presented in a way that makes all definitions visualizable and most results can be discovered geometrically (Moeller [18]).

Nearly every mathematics teacher is impressed with the potential for enhancement of teaching and problem solving by use of computer graphics. However, we are now accumulating experiences in attempts to realize that potential, and much of that experience is raising cautions about extravagant promises of easy or dramatic progress in student understanding and skill. First, we are discovering that, when it comes to graphs, "beauty is in the eye of the beholder"; many students find it very difficult to develop skill in interpreting graphs (Goldenberg *et al.*, 1988; Clement, 1985). Many students seem unable to resist the temptation to interpret graphs, like those showing projectile speed as a function of time, as concrete pictures of the physical situation which the function is modeling. For example, in a task asking students to sketch a graph of roller coaster speed over time, the most common response is a graph that simply copies the profile of the roller coaster track. Second, there are a number of pitfalls inherent in graphical computation that can lead students who have inadequate theoretical understanding to very serious incorrect conceptions (Demana and Waits, 1988). For instance, unfortunate choice of scales can lead to very similar pictures of functions with very different rules (see Figure 6). This example raises a question parallel to the concepts/skills controversy surrounding use of numerical calculators – what is the proper balance between use of the computer graphic tools and formal mathematical instruction that includes production of the same graphics by hand? A recent study by Dreyfus and Eisenberg [9] makes a first step toward understanding aspects of this situation, but there are very important research questions remaining.

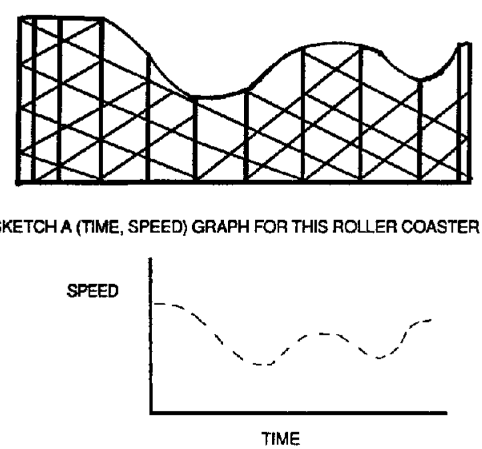
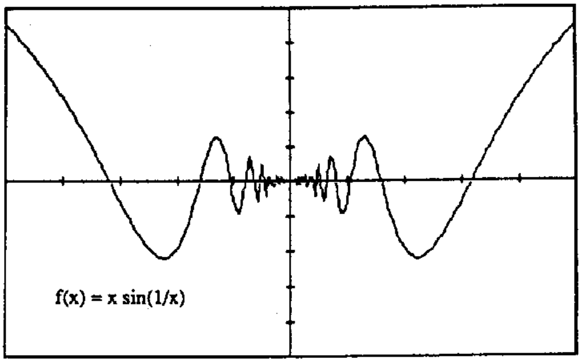
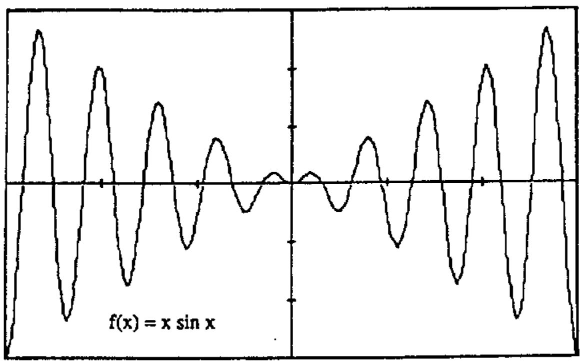


Fig. 5. A typical problem relating graphs to the events which they model.



[-0.5, 0.5] by [-0.5, 0.5].



[-30, 30] by [-30, 30].

Fig. 6. Graphs of x sin($1/x$) and x sin(x) made to appear similar by unfortunate choice of scales (Demana and Waits, 1988).

Summary

Computer graphics have been one of the most exciting contributions to mathematics education in this decade. They offer enormous promise for enhancing student understanding of important mathematical ideas and for providing alternative visual methods in mathematical problem solving. However, realization of this promise will require very careful research and development projects to overcome some difficult teaching/learning problems.

SYMBOLIC COMPUTATION AND MATHEMATICS EDUCATION

Mathematics educators have long been aware that computers are very useful for manipulation and storage of numerical data. However, for many of these same people the development of microcomputer programs that manipulate symbolic expressions according to the rules of algebra and calculus has been quite a surprise. Mathematicians have been using mainframe symbol manipulation programs for several decades, but muMath brought that power to 64K microcomputers in 1980, and there are now several other commercially available similar programs. Furthermore, the HP-28S computer/calculator makes a fairly impressive package of symbol manipulation programs available in a handheld device.

The general characteristics of computer symbol manipulation (or computer algebra) software include algebraic transformations and solution of equations involving polynomial, rational, and algebraic expressions – including elementary transcendental functions; matrix operations; calculation of derivatives and indefinite and definite integrals; calculation of power series expansions for given functions; summation of series; and solution of differential equations. In each case, the expressions involved can include numerical and/or literal parameters. The programs are generally quite fast, and they can deal with nearly every situation appearing in secondary school and early university mathematics. While some early versions used rather crude display formats, current packages present the results in forms that are identical to those of standard mathematical notation.

Since the symbol manipulation software has become available for microcomputers so recently, there is only a modest collection of research and development experiences with it. Proponents have speculated that there are at least three ways that its use could make significant impact on mathematics education. First, it seems clear that the software power extends the complexity of algebraic expressions that can be effectively handled at any level of instruction. Second, with computer assistance on routine symbol manipulation it seems quite possible to reorient instruction to focus on the conceptual understanding and procedural planning that remain essential in mathematical problem solving. Third, in much the same way that the Geometric Supposer or Cabri-Geometry tools facilitate exploratory learning, symbol manipulation utility programs can support rapid exploration of patterns in algebraic reasoning – leading to discovery of important general principles.

At this time there is little more than anecdotal or limited research evidence that any of the envisioned payoffs will follow from use of symbol manipulation software in mathematics education. Studies by Heid (1984, 1988), Palmiter (1986), and Judson (1988) have shown that use of symbol manipulation software in teaching calculus does permit greater emphasis on concept development and problem solving and that this change of priorities pays off in greater student understanding and skill in those aspects of the subject. In the Heid study students were taught very little about the traditional symbol manipulation rules until the last several weeks of the half-year course. Nonetheless, subsequent limited training in procedures – based on deeper and more confident understanding of fundamental ideas – was sufficient to produce skill levels equal to those of students who had spent an entire course with instruction that stressed skill acquisition.

In a study using muMath as a tool in elementary algebra Heid and Kunkle (1988) found that student problem solving abilities were enhanced by the new balance in use of instructional time. Similar results have been reported by Lesh (1985). Hosack (1988) reported a variety of informal studies using symbol manipulators in university mathematics courses, concluding that each use did indeed change the focus of student attention and emphasize the central ideas of each subject rather than procedural details.

With symbol manipulation utilities now available on hardware nearly as convenient as handheld calculators, it seems quite possible that we could make significant improvements in student understanding and problem solving by decreasing the required agenda of algebraic manipulative skills. Of course, there is reasonable concern that diminished command of traditional skills would have other damaging effects in advanced study. This is clearly an area of exciting opportunity, but very significant unanswered research questions.

Use of computer symbol manipulation or computer algebra systems as tools for learning about symbol manipulation itself is an almost totally unstudied area. But imagine the discoveries that students could make if they could call on an algebraic assistant to test the effects of various operations on a planned series of example expressions.

Summary

Software for symbolic reasoning in various strands of the mathematics curriculum is now available in sophisticated and easy to use form on a variety of widely used machines. Its potential for reshaping the content and teaching of various topics has been outlined in speculative papers and some curriculum experiments. However, the promise and potential problems that may result from that use are largely unknown at this time.

COMPUTER BASED MULTIPLE REPRESENTATIONS IN MATHEMATICS EDUCATION

The numerical, graphic, and symbol manipulation tools provided by computers each offer unique kinds of insight and power in mathematical teaching, learning, and problem solving. However, the feature of computers that has recently generated most excitement among mathematics educators is the ease of moving from one form of information representation to another as the user searches for conceptual understanding and problem solutions. When development projects make use of numerical or graphic data to teach about and solve problems involving variables and functions, they are really testing the hypothesis that multiple representations are helpful and that ability to translate an idea from one notation to another is an indicator of meaningful knowledge.

Of course, promised benefits from use of multiple representations or embodiments are not new in mathematics education. Textbooks and lectures have always relied on graphs to illuminate the properties of algebraic expressions and functions in calculus. Several attractive computer programs for teaching elementary mathematics are really just simulations of activities originally designed for use with physical materials like Dienes blocks. There are, however, several ways in which computer-based representations of mathematical ideas are unique and especially promising as instructional and problem solving tools.

First, computer representations of mathematical ideas and procedures can be made dynamic in ways that no text or chalkboard diagram can. Computer models of geometric transformations or of the changes in a function graph that correspond to parameter changes in a function rule do something that is very difficult to display otherwise. Second, the computer makes it possible to offer individual students an environment for work with representations that are flexible (like a set of Dienes blocks), but at the same time, constrained to give corrective feedback to each individual user whenever appropriate. As Fischer (in press) has noted

> The main difference between the mode of representation given by written symbols and by the computer is the last one's potentiality of performing operations by itself, whereas in the old mode humans had to act.

Third, while some multiple embodiment computer programs might be viewed as poor simulations of more appropriate tactile activity, it has been suggested that this 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. In this sense the computer plays a role as a kind of intermediate abstraction. Fourth, the versatility of computer graphics has made it possible to give entirely new kinds of representations for mathematics – representations 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 – not simply serving as heuristic sketches or "guess-and-test" calculations to get started on the path to more serious or closed form solutions. Recent work has shown the promise and some of the problems in each of these features of computer multiple representations.

Several of the programs described in earlier sections of this survey demonstrate the possibilities for dynamic multiple representations of mathematics in microworld environments which give users the kind of constrained flexibility that offers such promise. There are many others being tested all over the world. For instance, Klep and Gilissen [17] have designed a program to help students learn multiplication facts with the assistance of four models: jumps on a number line, $m \times n$ rectangular grids, bars like cuisenaire rods, and a set loop containing multiple copies of a given set. Students can move back and forth between models with ease, and the models are created in steps before the student's eyes. Lesh, Post, and Behr (1987) describe a similar model for representation of fraction concepts and facts, and Thompson [21] has created a Macintosh program in which students are invited to use the mouse to manipulate electronic Dienes blocks just as they might the actual materials.

Extending this kind of multiple representation manipulative support to instruction in algebra, Zehavi [24] devised a game called "Maxmix". Based on the simple task of choosing an operation that will maximize the result of combining two given numbers, the program leads students in levels from simple numerical data to points in a coordinate plane to tasks that require simultaneous consideration of three inequalities in two variables. Taizi and Zehavi (1985) created a related game called "Conquer the Plane" and they have used it with Maxmix in studies designed to see what difficulties students might have in working across such multiple representations. Tirza and Hershkowitz [23] have yet a third variation on this theme of using computer software to demonstrate links between different representations of mathematical information.

Each of the preceding examples involves use of the computer to present multiple embodiments of mathematical concepts and methods, where the representations are electronic versions of familiar instructional materials.

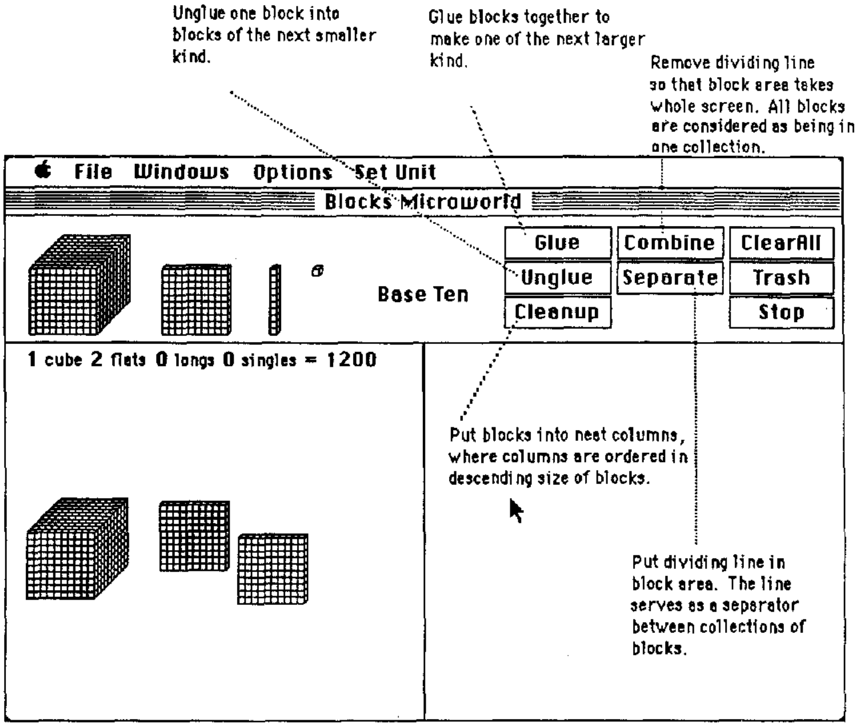


Fig. 7. A typical screen display from the Blocks microworld (Thompson [21]).

There are other development projects, however, in which something like 'creative iconography' is used to produce new kinds of representations. For instance, Feurzeig and his associates have experimented with use of a 'marble bag microworld' in which simple algebraic expressions and operations are represented symbolically, verbally, and iconically using marble bags for variables (Feurzeig, 1986; Roberts, Carter, Davis, and Feurzeig, 1987). Again, the idea is to help students make the transition from concrete to abstract reasoning through experience in an iconic world of intermediate abstraction.

Another very exciting example of representation by specially created computer graphic icons is the work by Tinker and Roberts (Tinker [22]; Roberts and Barclay, in press) in producing an integrated set of computer tools for modeling of dynamical systems. The system uses science laboratory interface software to collect experimental data and a spreadsheet program to display the data in numerical and graphic form. But the central conceptual tool in the modeling process is a program called STELLA in which the user creates schematic diagrams by arranging computer icons for

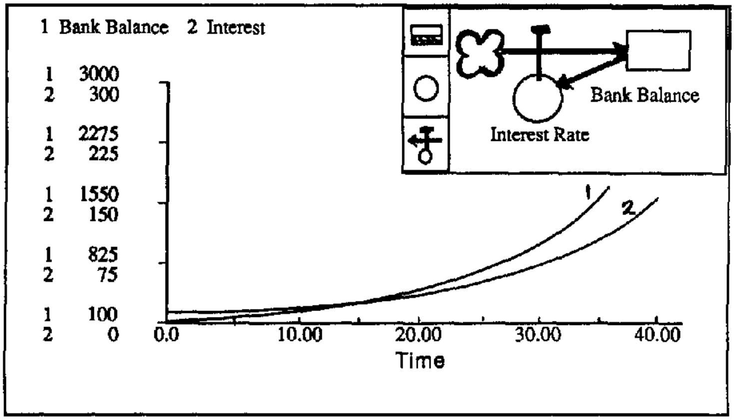


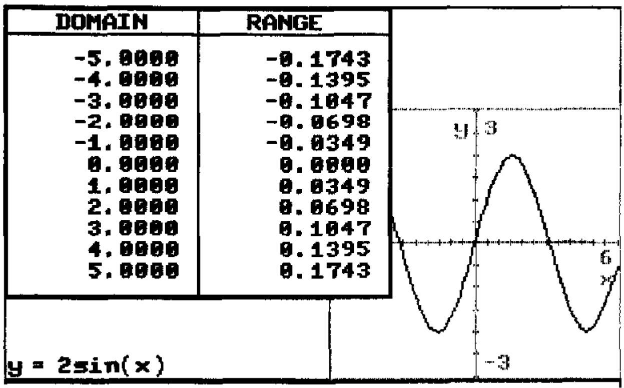
Fig. 8. Multiple representations of data and a model using the Stella modelling software (Roberts and Barclay, in press).

pipes, valves, and reservoirs to indicate rates of change and accumulation in systems that vary over time. Once the schematic is established and modeling data are entered, the computer solves the underlying differential equations and displays the behavior of the system. The fascinating aspect of this system is the way it supports user thought with a flexible graphic sketch pad. It shows the computer being used to enable students to think about and work effectively in situations of realistic complexity that would ordinarily be considered inaccessible at their level of mathematical *skill.*

The fact that Tinker and Roberts' modeling environment takes over much of the mathematical calculation, once a model is worked out by the user, illustrates another significant way in which computer based systems for multiple representation of mathematics are being applied. As Brolin and Greger (1987, p. 1) point out,

> With few exceptions, mathematical activities in school, high school, college, and at an introductory university level consist of symbol manipulation.
>
> (But) in the long run it will become impossible to continue to teach intelligent human beings the kind of skills which any dumb but adequately programmed computer can perform better and quicker.

To create a new more realistic world of mathematics, they designed a 'Mathematical Work Shop' that students could use for the routine tasks of problem solving so that their minds would be freed for those aspects of problem solving that require human intelligence. Brolin and Greger's Work Shop included programs for graphing, numerical tables, and solution of equation, inequality, and optimization problems. They then developed



DOMAIN	RANGE
-5.0000	-0.1743
-4.0000	-0.1395
-3.0000	-0.1047
-2.0000	-0.0698
-1.0000	-0.0349
0.0000	0.0000
1.0000	0.0349
2.0000	0.0698
3.0000	0.1047
4.0000	0.1395
5.0000	0.1743

Fig. 9. A typical screen display for multiple representation function tool software.

curriculum materials that would focus student attention on the mathematization process, on choice of appropriate tools, and interpretation of computed results. Here the multiple representation environment is used as a problem solving tool, and in the process it has profound implications for change in the kind of knowledge students must acquire from mathematics instruction. There is promise that, using such an approach, students will be able, as with modeling tools like STELLA, to gain access to much more challenging problem solving material. But such approaches run counter to strong beliefs of many mathematics teachers that the kinds of conceptual understanding that are still important can develop only out of extensive preliminary experience with hand calculations and that mathematical methods must be built on a sound foundation of logical derivation.

The consequences of using computer tools to change the skills/concepts/problem solving balance in mathematics curricula are probably the most important issue for research in technology applied to our field. There are some results beginning to appear that support the proposals for change. However, it is clear we have much to learn about most effective use of the computer tools and the new curricular organizations. We need to know more about the experiences which best develop student ability to move back and forth between representations (see, for example, Guin [15]; Dreyfus and Eisenberg [9]). And we also need further studies of effects from prototype curricula embodying the new style of mathematical work.

Summary

One of the brightest hopes for improvement in mathematics education by application of technology currently lies in the various applications of computer-based multiple linked representations for mathematical ideas and methods. Research and development projects are beginning to explore new curricular and instructional approaches that exploit this machine capability in various ways. It is clear that fundamental changes in traditional methods of mathematical teaching and learning are possible. But it is also clear that capitalizing on the advantages and minimizing the difficulties will require a great deal of thoughtful research and development work.

PROGRAMMING, INFORMATICS AND MATHEMATICS EDUCATION

While computers are well known as powerful machines for numerical, graphic, symbolic, and logical operations, the unique feature that enables all the power is the fact that computers are machines that can be programmed. In the first applications of computers to mathematics teaching and learning, students were almost always involved in writing programs. Other than the practical fact that in those early days any computer user had to write his or her own program to get a job done, programming activities were justified by the assumption that analyses required in writing a program would deepen student understanding of the underlying mathematics. In the past decade both the practical reality of computer use and beliefs about virtues of programming have changed. Now it is most common for computer users of all sorts to use a variety of special purpose program tools that require little more than entry of specific data and choice of procedural options from menus written in natural language. Furthermore, the search for evidence that programming experience influences mathematical behavior has not produced consistent or striking results. As a consequence, study and debate focused on programming and mathematics education have diminished substantially in the 1980's.

Some of the sharpest critical remarks about effects of programming have been directed at Logo, for which some dazzling initial promises were made (see example, Bender, 1987). But Papert (1987) replies that we should not ask only what Logo does to students, but what students can do with Logo. The technology theme group sessions at ICME VI revealed deep interest in Logo by mathematics educators from many different countries. Many investigators are still commited to exploring the broad potential of Logo for changing the mathematics learning environment and goals in fundamental ways (see, for example, Noss, 1988).

In a recent paper Blume and Schoen (1988) summarize the arguments suggesting that programming experience ought to have beneficial effects on mathematical problem solving abilities of students. They note that because of the nature of programming

We might expect programmers to (a) be more active and systematic in both the planning and the solution stages, (b) make more use of successive approximations to solutions, (c) make more use of variables and equations, and (d) be more likely to check for and correct errors in attempted solutions.

However, they conclude that

Despite the logical connection, however, studies to date have not provided strong evidence that programmers and nonprogrammers perform differently when solving mathematical problems. (p. 143)

Research by Blume and Schoen themselves found some evidence that programmers were more likely to use systematic trial approaches in problem solving and that they more frequently checked for and corrected errors in potential solutions. However, the other hypothesized behaviors were not observed more frequently among programmers than nonprogrammers.

While the pattern of results reported by Blume and Schoen has dampened enthusiasm for coupling programming and mathematics instruction, it seems premature to close the books on this issue. What many researchers have failed to find is a broad transfer effect from general programming activities to general problem solving behaviors. Far transfer effects are notoriously hard to find in any kind of educational research, and very few studies of any kind in mathematics education manage to make significant change in student problem solving behavior. Thus, it seems quite possible that there are benefits from programming which we have not detected in our quest for dramatic effects on difficult teaching/learning tasks like problem solving.

The ability to program a computer is an empowering skill that helps computer users reach beyond the constraints of packaged tool software. Furthermore, teaching of programming itself is a fairly new task in school, so it seems quite plausible that we don't yet know the most effective ways of teaching that skill, much less helping students make the connections between programming and mathematics.

Recent research on programming effects seems to have turned to more targetted approaches to make programming activities in a variety of languages pay dividends in mathematics learning. Kowszun and Higgo (1986), Thomas (1987), Ayers *et. al.* (1988), DeGraeve (1987), Capuzzo Dolcetta *et. al.* [6] and many others have recently reported work that takes this direction. It may very well turn out that programming can play a very positive role in mathematics education, but we have not yet done a very good job of realizing that promise.

Of course, programming is only one aspect of the discipline of computer science or informatics. The debate over effects of programming on mathematical learning looks at the interplay of the two disciplines from only one direction. Engel (1983), Maurer (1983, 1984), Ralston (1985) and many others have pointed out that the crucial step in applying computation power to a problem is the design of a suitable algorithm to guide the information processing operations. Thus it is important for students of mathematics to learn effective algorithms for important mathematical problems and to develop a more general ability to create algorithmic solutions to novel problems. As a British conference report put the challenge, "The study of algorithms and procedures represents both an extension of mathematics and a new way to view the current school curriculum" (Kowszun and Higgo, 1986). Thus such debates as the value of programming to mathematics learning or the virtues of various programming languages obscure a more fundamental challenge – revising our approaches to mathematics so that we prepare our students to apply the algorithmic methods which are essential in the use of computers.

The questions concerning programming and algorithmic methods in mathematics are really only specific dimensions of a broader problem – coordinating the school curricula in informatics and mathematics to the best advantage of each. While there has been some speculative discussion of how this might be accomplished (Ralston, 1985; Bottino, Forcheri, Furinghetti, and Molfino [5]), it stands right now as largely an unsolved problem. From the mathematical side we must decide which topics from discrete mathematics (logic, difference equations, induction/recursion, finite probability, etc.) should, because of their value to informatics education and to understanding of powerful computer methods in mathematics, be included in school and university curricula. At the present time discussion and curriculum experimentation seems focused on upper secondary and early university study and on the (quite possibly false) competition for curriculum priority between discrete and continuous mathematics.

Summary

It is natural to believe that acquisition of skill in computer programming will develop habits of mind that will be helpful in various aspects of learning and doing mathematics. While research has yet to find convincing evidence of any such broad transfer effects, there is active but more focused interest in finding ways to make this interplay effective. Development of mathematics curricula that emphasize algorithmic methods would serve students well when they turn to computers for problem solving help, but thus far this emphasis on algorithmic methods seems not to have made major impact on school or university mathematics.

ARTIFICIAL INTELLIGENCE, EXPERT SYSTEMS, AND TUTORS

Each preceding section of this survey describes ways that electronic information technology influences mathematics education by providing tools to assist with mathematical procedures – arithmetic calculations, graphic displays, symbol manipulation, and execution of algorithmic processes. Availability of those mathematical tools suggests changes in the goals of school and university curricula and in traditional patterns of teaching/learning activity. But one of the very active dimensions of informatics research is exploring ways that computers can be programmed to exhibit 'behavior' that simulates human information processing. There are a number of projects in mathematics education that are attempting to capitalize on this computer capability to design programs that act, in various ways, like teachers.

The initial efforts to educate computers as teachers were of two main types. The first were variations on the electronic flash card theme – drill and practice programs in which the computer posed the problems and gave students immediate feedback on their performance. These programs are still very popular in schools and, for the purpose of sharpening necessary skills, they are apparently successful. We are beginning to see an array of programs that provide drill and practice in very clever settings and that provoke higher level strategic thinking as well as routines. There are impressive examples in the work of the Shell Centre, Dugdale and Kibbey, and Taizi and Zehavi – to mention only three sources.

The second way in which computers have been used to simulate teaching is as an electronic medium for programmed instruction. In the quest for a computer-based course that could operate independent of any teacher or textbook, developers have often translated programmed text to the computer screen, devoting considerable energy to the problem of teaching the computer to respond to student entries in intelligent ways. As with drill and practice, there are examples of such programmed electronic courses still in use and still being developed. But it seems fair to say that interest has largely shifted from 'stand-alone' instructional systems that deliver, assess, and manage all aspects of education to development of more focused uses for computer tutors as only part of the teaching/learning environment.

The most interesting new work along these lines is the production of *intelligent tutors* for various mathematical domains. Schoenfeld (1988, p. 7) describes the goal of this work:

> The system would present information to the student; the student would work practice problems; the system could speed the student along when her work was going well, but could also diagnose the student's mistakes and help when things went wrong; and it could answer the student's questions on a wide range of related issues.

He goes on to point out that

> In order for any tutor (machine or human) to succeed at this, it has to be (a) expert at the subject matter, (b) pretty good at figuring out what's going on in the student's head, and (c) pretty good at teaching (i.e. have accessible a wide range of teaching strategies).

As mentioned earlier in this survey, there has been remarkable progress toward development of computer tools that are 'expert' at various aspects of mathematics – from arithmetic and algebra to geometry and calculus. Research in cognitive science is yielding a fairly comprehensive description of common conceptual and procedural errors in those topic areas. But combination of those ingredients into a flexible, responsive instructional tutor that interacts with students in depth is clearly a very difficult problem. Work on tutors in arithmetic (Floyd, Hennessy, and O'Shea [11]; Ohlsson and Resnick [19]), algebra (Anderson, Boyle, and Reiser, 1985; McArthur, Stasz, and Hotta, 1987), geometry and proof (Bell, 1987; Flake [10]; Barz and Holland [2]) and calculus (Suppes, Ager, and Berg [20]) is well underway. There are some preliminary indications that those tutors provide very effective adjuncts to (and in some cases substitutes for) regular teacher directed instruction.

The prototype tutors provide some very interesting learning environments. For instance, the algebra tutors commonly display student work or model solutions in schematic diagrams that show how reasoning steps are assembled into arguments. Then they permit student inspection and modification of any stage in the process, with explanations and corrective guidance available in varying degrees at each stage. The geometry tutors provide similar diagrammatic displays and corrective feedback as students assemble constructions or proofs. The most recent attraction in this line of work is the promise of combining computer tutors with video-disk technology to give a visually richer tutoring environment (Terada, 1987; Terada, Hirose, and Handa, 1985). There are some stunning examples of what this combination could produce, and some others that provide little more than teacher lectures recorded on video-disks.

While these early efforts at intelligent tutors have often been criticized for

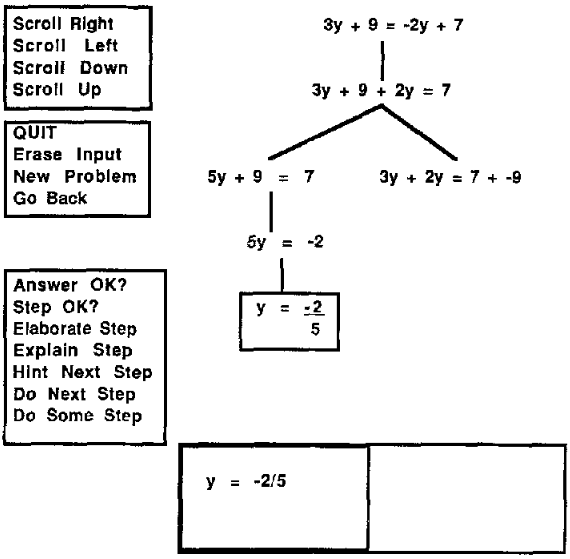


Fig. 10. A typical screen display for an algebra tutor (McArthur *et al.*, 1987).

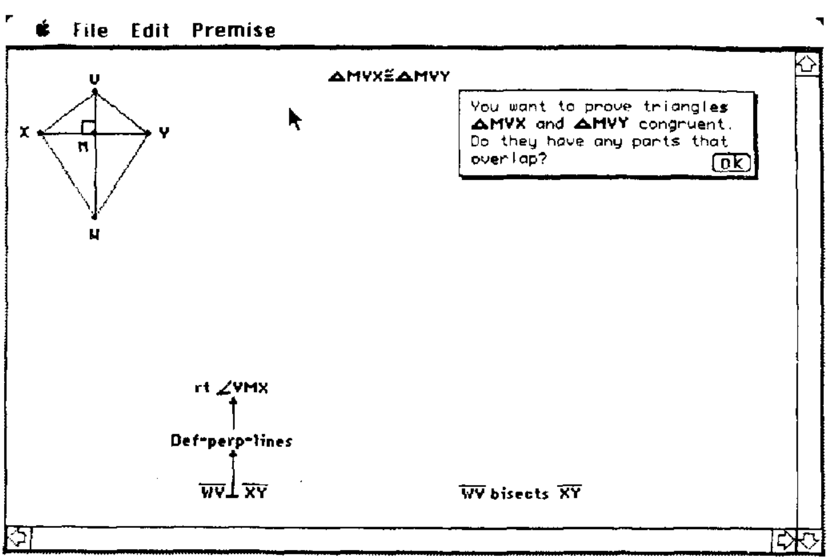


Fig. 11. A typical screen display for a geometry tutor (Anderson *et al.*, 1985).

focusing on teaching the very skills that computers make obsolete, the design effort has a number of very productive consequences for mathematics education. First, to design a good tutor one must make an intensive study of the ways that students process instructional information and the most likely spots for difficulty in learning and problem solving. Second, design of appropriate computer-based instructional methods is generally based on in-depth analysis of the strategies of effective teacher-tutors. Examination of traditional teaching from this new perspective often yields productive insights into research on teaching.

Summary

There is now a small but growing body of research in mathematics education seeking to combine the insights of cognitive science and artificial intelligence to produce 'expert' computer tutoring systems for various subjects. The most impressive of the current examples run on rather expensive machines and deal effectively with only limited aspects of mathematics. The dialogue capabilities of the systems are fairly limited. However, there is quite reasonable hope that steady progress can be made along this front, providing yet another way that computing can influence the shape of mathematics education.

CONCLUSIONS AND PROSPECTS

The array of computer applications described in this survey and the countless projects which cannot be mentioned due to limits of space and time constitute the single most powerful force for change in school and university mathematics education today and in the near future. The potential in using technology to extend the range of human mathematical learning and problem solving is only beginning to be tapped by research and development projects, much less in the day-to-day life of mathematics classrooms. While some may choose to wait until a clearer picture of the 'best' response emerges, the situation right now offers impressive opportunities for progress.

Effective use of computers for instruction can permit the kinds of teaching/learning environments that most teachers long for, while they struggle with the constraints of traditional classroom and curricular conditions. Revision of curricular goals to acknowledge that computers and other electronic information technologies are now standard tools for problem-solving and decision-making in science, business, government, and industry will lead to significant change in what we ask and empower students to learn. There are many important questions to be answered, but we really have no choice except to tackle those questions and to bring school and university mathematics into the electronic information age for which we are ostensibly preparing our students.

NOTES

Much of the information used in preparing this survey paper was provided by ICME VI participants in papers written for that meeting but not yet published in journal articles or books. The following reference notes indicate authors and titles of such papers, with an institutional affiliation that should be of some help to readers who want to get more information about a particular project. It seems likely that many of the ideas and projects cited here will be reported soon in regular publication channels.

[1] Artigue, M., Gautheron, V., and Sertenac, P., 'A qualitative study of differential equations: results from some experiments with microcomputers' [IREM Paris 7, 2 Pl. Tussieu, 75005 Paris, France].

[2] Barz, W. and Holland, G., 'Intelligent tutorial systems for geometric construction and proof problems' [Institut fur Didaktik der Mathematik, Universitat Giessen, Karl-Gloeckner-Strasse 21 C, D-6300 Giessen, Federal Republic of Germany].

[3] Bellemain, F., 'Le Projet cabri-geometre: Elaboration d'un logiciel d'aide a la construction et la transformation de figures pour la resolution de problem de geometrie Euclidienne' [Laboratoire de Structures Discretes et Didactique, IMAG, BP68, 38402 Saint Martin D'Heres Cedex, France].

[4] Bjork, L. and Brolin, H., 'The role of a mathematical workshop in the teaching of mathematics' [Sunnerboskolan, Axel Roothsgatan 13, S-34100 Ljungby, Sweden and Department of Teacher Training, University of Uppsala, Box 2136, S-75002 Uppsala, Sweden].

[5] Bottino, R. M., Forcheri, P., Furinghetti, F., and Molfino, M. T., 'Integration between mathematics and computer science: Reflections and realizations' [Istituto di Mathematica, Applicata del C. N. R., Via L. B. Alberti 4 – 16132 Genova, Italy].

[6] Capuzzo Dolcetta, I., Emmer, M., Falcone, M., and Finzi Vita, S., 'The impact of new technologies in teaching calculus: A report of an experience' [Dipartimento di Mathematica, Universita di Roma "La Sapienza", P. Aldo Moro, 2 – 00185 Roma, Italy].

[7] Cesar, J., '3D-extended logo: The sea-turtle and perspective' [2 rue sur la cote, Arbouans, 25400 Audincourt, France].

[8] Cornu, B., 'The computer and didactics: Two tools for the teaching of mathematics' [Institut Fourier-Equipe de Recherche en Didactique des Mathematiques et de l'Informatique, Universite Grenoble 1 – France].

[9] Dreyfus, T. and Eisenberg, T., 'On visualizing function transformations' [The Weizmann Institute of Science, Rehovot 76100, Israel].

[10] Flake, J. L., 'Using computers for developing proof skills' [Department of Curriculum and Instruction, Florida State University, Tallahassee, FL, USA].

[11] Floyd, A., Hennessy, S., and O'Shea, T., 'An intelligent computer tutor for arithmetic' [School of Education, Open University, Milton Keynes, United Kingdom].

[12] Fraser, R., 'Role of computers for application and modelling: The ITMA approach' [Shell Centre for Mathematical Education, University of Nottingham, United Kingdom].

[13] Fraser, R., Burkhardt, H., Coupland, J., Phillips, R., Pimm, D., and Ridgway, J., 'Learning activities and classroom roles' [Shell Centre for Mathematical Education, University of Nottingham, United Kingdom].

[14] Friel, S. N. and Russell, S. J., 'Used numbers: Collecting and analyzing real data' [Technical Education Research Centers, 1696 Massachusetts Avenue, Cambridge, MA, USA].

[15] Guin, D., 'Logo, un outil pour mettre en jeu les differents registres de la notion de fonction' [I.R.E.M. Strasbourg, 10, rue du General Zimmer, 67084 Strasbourg Cedex, France].

[16] Klep, J., 'The problem of controlling complex learning processes' [National Institute for Curriculum Development, Enschede, The Netherlands].

[17] Klep, J. and Gilissen, L., 'Computer help in learning the multiplication tables' [National Institute for Curriculum Development, Enschede, The Netherlands].

[18] Moeller, H., 'Elementary analysis with microcomputers' [Heinrich-Behnke-Seminar fur Didaktik der Mathematik, Universitat Munster, Einsteinstrasse 62, D-4400 Munster, Federal Republic of Germany].

[19] Ohlsson, S. and Resnick, L., 'Intelligent tutors for elementary and middle school mathematics' [Learning Research and Development Center, University of Pittsburgh, Pittsburgh, PA, USA].

[20] Suppes, P., Ager, T., and Berg, P., 'Applications of computer technology to pre-college calculus' [Stanford University, Stanford, CA, USA].

[21] Thompson, P. W., ' "Direct engagement" software for teaching mathematics' [Department of Mathematics, Illinois State University, Normal, IL, USA].

[22] Tinker, R. F., 'Educational technology and the future of mathematics and science education' [Technical Education Research Centers, 1696 Massachusetts Avenue, Cambridge, MA, USA]

[23] Tirza, H. and Hershkowitz, R., 'Exploring the coordinate system using microcomputer software' [The Weizmann Institute of Science, Rehovot 76100, Israel].

[24] Zehavi, N., 'The development of mathematics software: An interactive process' [The Department of Science Teaching, The Weizmann Institute of Science, Rehovot 76100 Israel].

REFERENCES

Anderson, J. R., Boyle, C., and Reiser, B.: 1985, 'Intelligent tutoring system', *Science* **228**, 456–462.

Ayers, T., Davis, G., Dubinsky, E., and Lewin, P.: 1988, 'Computer experiences in learning composition of functions', *Journal for Research in Mathematics Education* **19**, 246–259.

Bell, M. S.: 1987, 'Microcomputer-based courses for school geometry', in I. Wirszup and R. Streit (eds.), *Developments in School Mathematics Education Around the World*, National Council of Teachers of Mathematics, Reston, VA, pp. 604–623.

Bender, P.: 1987, 'Kritik der logo-philosophie', *Journal fur Mathematik-Didaktik* **8**, 3–103.

Bloom, L. M., Comber, G. A., and Cross, J. M.: 1986, 'Use of the microcomputer to teach the transformational approach to graphing functions', *International Journal of Mathematical Education in Science and Technology* **17**, 115–123.

Blume, G. W. and Schoen, H. L.: 1988, 'Mathematical problem-solving performance of eighth-grade programmers and nonprogrammers', *Journal for Research in Mathematics Education* **19**, 142–156.

Brolin, H.: 1987, 'Mathematics in Swedish schools', in I. Wirszup and R. Streit (eds.), *Developments in School Mathematics Education Around the World*, National Council of Teachers of Mathematics, Reston, VA, pp. 130–155.

Brolin, H. and Greger, K.: 1987, *Computers and the Teaching of Mathematics*, Uppsala University, Department of Teacher Training, Uppsala.

Campbell, P. F.: 1987, *Measuring Distance: Children's Use of Number and Unit* (Final Report), Department of Curriculum & Instruction, University of Maryland, College Park, MD.

Clement, J.: 1985, 'Misconceptions in graphing', *Proceedings of the Ninth International Conference for the Psychology of Mathematics Education*, pp. 369–375.

Clements, D. H.: 1987, 'Longitudinal study of the effects of logo programming on cognitive abilities and achievement', *Journal of Educational Computing Research* **3**, 73–94.

Corbitt, M. K.: 1985, 'The impact of computing technology on school mathematics: Report of an NCTM conference', *Arithmetic Teacher* **32**, 14–19.

Danby, J. M. A.: 1988, 'Computer applications in differential equations', in D. A. Smith, G. J. Porter, L. C. Leinbach, and R. H. Wenger (eds.), *Computers and Mathematics: The Use of Computers in Undergraduate Instruction*, The Mathematical Association of America, Washington, DC, pp. 35–42.

DeGraeve, R.: 1987, 'Logo au cp et au cel', *Revue de Mathematiques Sciences et Technologie pour les Maitres de l'Enseignement Elementaire et Pre-elementaire* **41**, 37–53.

Demana, F. and Leitzel, J.: 1988, 'Establishing fundamental concepts through numerical problem solving', in A. F. Coxford and A. P. Shulte (eds.), *The Ideas of Algebra K-12*, 1988 Yearbook of the National Council of Teachers of Mathematics, Reston, VA, pp. 61–68.

Demana, F. and Waits, B. K.: 1988, 'Pitfalls in graphic computation, or why a single graph isn't enough', *The College Mathematics Journal* **19**, 177–183.

Dugdale, S.: 1987, 'Pathfinder: A microcomputer experience in interpreting graphs', *Journal of Educational Technology Systems* **15**, 259–280.

Dugdale, S.: 1984, 'Computers: Applications unlimited', in V. P. Hansen and M. J. Zweng (eds.), *Computers in Mathematics Education*, 1984 Yearbook of the National Council of Teachers of Mathematics, Reston, VA, pp. 82–88.

Engel, A.: 1983, 'The impact of algorithms in mathematics teaching', *Proceedings of the Fourth International Congress on Mathematics Education*, Birkhauser, Boston, MA, pp. 312–330.

Feurzeig, W.: 1986, 'Algebra slaves and agents in a logo-based mathematics curriculum', *Instructional Science* **14**, 229–254.

Fey, J. T. (ed.): 1984, *Computing and Mathematics: The Impact on Secondary School Curricula*, National Council of Teachers of Mathematics, Reston, VA.

Fey, J. T.: 1989, 'School algebra for the year 2000', in C. Kieran and S. Wagner (eds.), *Research Issues in the Learning and Teaching of Algebra*, National Council of Teachers of Mathematics, Reston, VA.

Fischer, R.: in press, 'The "human factor" in pure and applied mathematics', in M. S. Arora, A. Rogerson, and F. Mina (eds.), *Mathematics Education into the 21st Century*.

Fraser, R.: 1986, June, 'Learning activities and classroom roles: Communication', Paper presented at the Conference on Enhancing Mathematics Teaching with Microcomputers, Coventry Polytechnic, United Kingdom.

Gaulin, C. and Puchalska, E.: 1987, 'Coded graphical representations: A valuable but neglected means of communicating spatial information in geometry', in I. Wirszup and R. Streit (eds.), *Developments in School Mathematics Education Around the World*, National Council of Teachers of Mathematics, Reston, VA, pp. 514–539.

Goldenberg, E. P., Harvey, W., Lewis, P., Umiker, R. J., West, J., and Zodhiates, P.: 1988, *Mathematical, Technical, and Pedagogical Challenges in the Graphical Representation of Functions* Education Development Center, Newton, MA.

Heid, M. K.: 1984, *Resequencing Skills and Concepts in Applied Calculus Using the Computer as a Tool*, Doctoral dissertation, University of Maryland, College Park, MD.

Heid, M. K.: 1988, 'Resequencing skills and concepts in applied calculus using the computer as a tool', *Journal for Research in Mathematics Education* **19**, 3–25.

Heid, M. K. and Kunkle, D.: 1988, 'Computer-generated tables: Tools for concept development in elementary algebra', in A. Coxford and A. Shulte (eds.), *The Ideas of Algebra, K-12*, 1988 Yearbook of the National Council of Teachers of Mathematics, Reston, VA, pp. 170–177.

Hembree, R. and Dessart, D.: 1986, 'Effects of hand-held calculators in precollege mathematics education: A meta-analysis', *Journal for Research in Mathematics Education* **17**, 83–89.

Hosack, J.: 1988, 'Computer algebra systems', in D. A. Smith, G. J. Porter, L. C. Leinbach, and R. H. Wenger (eds.), *Computers and Mathematics: The Use of Computers in Undergraduate Instruction*, The Mathematical Association of America, Washington, DC, pp. 35–42.

Judson, P. T.: 1988, *Effects of Modified Sequencing of Skills and Applications in Introductory Calculus*, Doctoral dissertation, University of Texas at Austin, TX.

Kowszun, J. and Higgo, J.: 1986, 'Algorithms and programming in the secondary mathematics curriculum', *Mathematics in School* **15**, 31–33.

Kramer, E., Hadas, N., and Hershkowitz, R.: 1986, 'Geometrical constructions and the microcomputer', *Proceedings of the Tenth International Conference for the Psychology of Mathematics Education*, pp. 105–110.

Lesh, R.: 1985, April, 'The evolution of problem representations in the presence of powerful conceptual amplifiers', Paper presented at the American Educational Research Association meeting, Washington, DC.

Lesh, R., Post, T. R., and Behr, M.: 1987, 'Dienes revisited: Multiple embodiments in computer environments', in I. Wirszup and R. Streit (eds.), *Developments in School Mathematics Education Around the World*, National Council of Teachers of Mathematics, Reston, VA, pp. 647–680.

McArthur, D., Stasz, C., and Hotta, J. Y.: 1987, 'Learning problem-solving skills in algebra', *Journal of Educational Technology Systems* **15**, 303–324.

Maurer, S. B.: 1983, 'The effects of a new college mathematics curriculum on high school mathematics', in A. Ralston and G. S. Young (eds.), *The Future of College Mathematics*, Springer-Verlag, New York, pp. 153–173.

Maurer, S. B.: 1984, 'Two meanings of algorithmic mathematics', *The Mathematics Teacher* **77**, 430–435.

Meissner, H.: 1987, 'Schulerstrategien bei einem taschenrechnerspiel', *Journal fur Mathematik-Didaktik* **8**, 105–128.

Mohyla, J. (ed.): 1985, *The Role of Technology. Theme Group 3 Report, ICME 5*, South Australian College of Advanced Education, Adelaide.

Noss, R.: 1987, 'Children's learning of geometrical concepts through logo', *Journal for Research in Mathematics Education* **18**, 343–362.

Noss, R.: 1988, 'The computer as a cultural influence in mathematical learning', *Educational Studies in Mathematics* **19**, 251–268.

Orzech, M.: 1988, 'Using computers in teaching linear algebra', in D. A. Smith, G. J. Porter, L. C. Leinbach, and R. H. Wenger (eds.), *Computers and Mathematics: The Use of Computers in Undergraduate Instruction*, The Mathematical Association of America. Washington, DC, pp. 63–68.

Palmiter, J. R.: 1986, *The Impact of a Computer Algebra System on College Calculus*, Doctoral dissertation, The Ohio State University, OH.

Papert, S.: 1987, 'Computer criticism vs. technocentric thinking', *Educational Researcher* **16**, 22–30.

Ralston, A.: 1985, 'The really new college mathematics and its impact on the high school curriculum', in C. R. Hirsch and M. J. Zweng (eds.), *The Secondary School Mathematics Curriculum*, 1985 Yearbook of the National Council of Teachers of Mathematics, Reston, VA, pp. 29–42.

Roberts, N., Carter, R., Davis, F., and Feurzeig, W.: 1987, *Power Tools for Algebra Problem Solving* (Final Report), Lesley College and Bolt, Beranek, and Newman, Cambridge, MA.

Roberts, N. and Barclay, T.: in press, 'Teaching model building to high school students: Theory and reality', *Journal of Computers in Mathematics and Science Teaching.*

Sandefur, J. T., Jr.: 1985, 'Discrete mathematics: A unified approach', in C. R. Hirsch and M. J. Zweng (eds.), *The Secondary School Mathematics Curriculum*, 1985 Yearbook of the National Council of Teachers of Mathematics, Reston, VA, pp. 90–106.

Sandefur, J. T., Jr. and Vogt, A.: 1988, 'Finite differences: A computer-based alternative to calculus', in D. A. Smith, G. J. Porter, L. C. Leinbach, and R. H. Wenger (eds.), *Computers and Mathematics: The Use of Computers in Undergraduate Instruction*, The Mathematical Association of America, Washington, DC, pp. 47–54.

Scheaffer, R. L.: 1987, 'Statistics and probability in the school mathematics curriculum: A review of the ASA/NCTM/NSF quantitative literacy program', in I. Wirszup and R. Streit (eds.), *Development in School Mathematics Education Around the World*, National Council of Teachers of Mathematics, Reston, VA, pp. 380–396.

Schoenfeld, A. H.: 1988, 'Uses of computers in mathematics instruction', in D. A. Smith, G. J. Porter, L. C. Leinbach, and R. H. Wenger (eds.), *Computers and Mathematics: The Use of Computers in Undergraduate Instruction*, The Mathematical Association of America, Washington, DC, pp. 1–13.

Schwartz, J. L.: 1987, 'The geometric supposer: Using microcomputers to restore invention to the learning of mathematics', in I. Wirszup and R. Streit (eds.), *Developments in School Mathematics Education Around the World*, National Council of Teachers of Mathematics, Reston, VA, pp. 623–636.

School Mathematics: New Ideas with Computers: 1987, National Institute for Educational Research, Tokyo.

Swets, J., Rubin, A., and Feurzeig, W.: 1987, *Cognition, Computers, and Statistics: Software Tools for Curriculum Design*, Bolt, Beranek, and Newman, Cambridge, MA.

Taizi, N. and Zehavi, N.: 1985, 'Microcomputer-assisted sophistication in the use of mathematical knowledge', *Proceedings of the Ninth International Conference on the Psychology of Mathematics Education*, pp. 101–104.

Tall, D.: 1985, 'Visualizing calculus concepts using a computer', in *The Influence of Computers and Informatics on Mathematics and its Teaching*, Institut de Recherches sur l'Enseignement des Mathematiques, Strasbourg, pp. 303–308.

Terada, F.: 1987, 'Learning mathematics through modern media and THE system', in I. Wirszup and R. Streit (eds.), *Developments in School Mathematics Education Around the World*, National Council of Teachers of Mathematics, Reston, VA, pp. 576–586.

Terada, F., Hirose, K., and Handa, T.: 1985, 'Towards a self-paced learning support system', in K. Duncan and D. Harris (eds.), *Computers in Education*, Elsevier Science Publishers, Amsterdam, pp. 39–44.

Thomas, M. O. J.: 1987, 'Algebra with the aid of a computer', *Mathematics in School* **16**, 36–40.

Winkelmann, B.: 1984, 'The impact of the computer on the teaching of analysis', *International Journal for Mathematics Education in Science and Technology* **15**, 675–689.

Wynands, A.: 1984, 'Rechenfertigkeit und taschenrechner – ergebnisse einer dreijahrigen untersuchung in den klassen 7–9', *Journal fur Mathematik-Didaktik* **5**, 3–32.

Yerushalmy, M. and Houde, R. A.: 1986, 'The geometric supposer: Promoting thinking and learning', *Mathematics Teacher* **79**, 418–422.

The author was assisted in collection and analysis of information for this survey by contributions from members of the theme group T2 organizing panel for ICME VI, organizers of the theme sub-groups, and many other individuals who contributed papers and insights in response to my requests. Special thanks are due to Mr. David Dyer who provided extensive help in the literature search and organization of what proved to be a small mountain of material related to the subject.

Research for this paper was supported in part by the United States National Science Foundation under NSF award no. MDR 87-51500. Any opinions or findings, conclusions, or recommendations expressed herein are those of the author and do not necessarily reflect the views of the National Science Foundation.

Department of Curriculum and Instruction and Mathematics,
University of Maryland,
College Park, MD,
U.S.A.