Algebra in a Computer Environment

David Tall
Mathematics Education Research Centre,
Warwick University
Coventry CV4 7AL, UK

This reaction has been commissioned by the PME XI organising committee to consider the contribution of each paper grouped under this heading, to seek common threads, to formulate major questions that still need to be answered and to look for indications in the papers as to how these questions might be tackled. The task is a daunting one. It is rather like attempting to put together a jigsaw puzzle whose pieces were not created to fit together in a master plan, each with a life of its own. It is a problem-solving activity and I shall approach it in a problem-solving spirit. In doing so I should like to acknowledge the help given me by Michael Thomas in formulating this reaction.

1. The contributions of the papers to the research area

The papers grouped under "algebra in a computer environment" range widely from initial ideas in the subject to the graphical representation of algebraic functions, and some expand the domain to more general functions and analytic relationships between variables and their rate of change in "feed-back systems". Although these would not all be classified *mathematically* within algebra, they *cognitively* embrace algebra concepts, beginning with the translation from real world problems to algebraic notation, with its surface syntax and underlying semantic structure, linking with relationships to other representational systems.

The papers also represent very different stages in the research process which are fruitfully considered from a problem-solving viewpoint, passing through various phases after the style discussed by Mason *et al.* (1982). An initial **entry** phase gathers together what is known, what one wants to know, and what tools one might assemble in preparation for the **attack** phase where the empirical work is done. This may result in an impasse or a significant gain, when it becomes appropriate to **review** and refine what has been achieved before either re-entering the problem for a different attack, or ing the work in new areas through a new a spiral of entry, attack and review.

Some of the papers have completed a full research cycle, others describe only part of a longer span, for instance, the entry phase to new research, reviewing the literature from earlier phases, proposing theories and setting out plans of attack.

Boileau *et al,* are beginning a new phase of attack in **"La pensée algorithmique dans l'initiation à l'algèbre"**. They propose to start the study of algebra with activities that are "both significant and motivating to the student", "coding problems ... relating to the students' prior experience" by

Published in *Proceedings of the Eleventh International Conference of P.M.E.*,
Montreal, I, 262–271 (1987).

providing a “tailor-made programming language which will serve as an intermediate representation ... between the problem to be solved and the final coding.” They formulate some characteristics of the environment but stop short of giving information as to the state of the development of the system or any empirical testing. Their distinction between *the syntactic*, *internal semantic* and *external semantic* aspects of algebra is one which may prove a useful link with other papers.

In **“believing is seeing: how preconceptions influence the perception of graphs**”, Goldenberg begins an entry phase, based on experiences using computer software and preliminary observations with two “bright, successful, second year algebra students”. He leads into a discussion of “how perceptual illusions and shifts of attention from one feature to another obscure some of what educational use of graphs is supposed to illucidate”, particularly where the representation lacks familiar perceptual clues, thus raising some concern as to the efficacy of certain aspects of multiple linked representations.

Thompson & Thompson introduce some significant new software in **“computer representations of structure in algebra”**, linking an algebraic expression of its tree structure allowing free mixing of numbers and letters. They have made an initial empirical attack with a week’s instruction/exploration of the software with eight seventh-grade students. They report that the students “felt no discomfort when letters were first introduced in to-be-transformed expressions” and that, after an initial period of experimentation, errors due to inattention to structure were infrequent.

Judah Schwartz also has a reputation for producing innovative software and his paper on **“the representation of function in the algebraic proposer”** is no exception in this respect. The original proposal had hoped to include empirical research with 12 college freshmen, but, in the event, the paper is restricted to a presentation and discussion of the software only, giving a tantalizing glimpse of the possibilities of providing a word problem an algebraic description and interrelating it with graphical and numerical representations.

Dreyfus and Eisenberg present a complete research cycle **“on the deep structure of functions”**, entering with a theoretical framework for analysing aspects of the function concept, empirical knowledge of student misconceptions, and a constructivist approach to abstraction using computer microworlds. They hypothesise that the understanding of the relationship between the algebraic and graphical representation of a function is facilitated by using a specific piece of software and that this can be improved by providing structured activities for the students. One group of eight students worked in a highly structured teaching environment whilst a second group were allowed to explore freely. A pre- and post-test revealed a significant improvement by both groups on “non-standard” questions, relating to shifting and stretching transformations on graphs, but the difference between groups was not significant.

In **“Dienes revisited: multiple embodiments in computer environments”**, Lesh & Herre report part of a major on-going project of research and curriculum development which reveals “significant ways that computer-based instruction can encourage teachers and students to make greater use of activities with concrete materials ... at the same time … implementing some of the best instructional strategies associated with mathematics laboratories”. They discuss a symbol-manipulator/function plotter called *SAM* which provides direct links between algebraic manipulation on equations and the graphical representations of the functions on each side of the equals sign. The general questions raised are broad and important but the page restriction regrettably leaves no room to report empirical results.

Zehavi et *al* cover a complete research cycle in **“the effects of microcomputer software on intuitive understanding of graphs and quantitative relationships”**. They describes a new piece of software, “*Dots and Rules*”, designed to help intuitions on graphs of linear relationships, tested using pupils of “average ability”, in three experimental classes compared with three control classes, selected from similar schools. Tests were given immediately after the treatment and eight months later and showed that “although the software seems to have been only moderately effective, retention of what was learnt was good”. “The results indicate that software of this kind can be effective in achieving its main goal – creating intuitive readiness for future concepts.”

Two papers look at the role of programming in Logo and its relationship to ‘paper and pencil’ algebra. Sutherland outlines the preliminary results of a three year case study on “... **the use and understanding of algebra-related concepts within a Logo environment”**. She reports that “analysis of the data indicates that most pupils do not naturally choose to use variables in their Logo programming, although with teacher intervention it is possible to find motivating problems which provoke pupils to use variables”. Under these circumstances there is evidence that “pupils can use their Logo derived understanding in an algebra context”.

In **“using micro-computer assisted problem-solving to explore the concept or literal symbols – a follow-up study”,** Nelson interviewed three “average ability students” a year after a study in which they had been “taught to use Logo to solve problems involving number sentences, rectangles and recursion”. They remembered most of the Logo commands used a year before, though none recalled the MAKE command for variables and “were able to use literal symbols to represent missing dimensions of rectangles when writing expressions for area”. The author concludes that “microcomputer oriented problem solving has a long-term effect on the concept of literal symbols”.

Two other papers beginning new entry phases of research pass beyond algebra into concepts linking variables and their rates of change. In **"Un système d'apprentissage de l'abstraction par représentation graphique",** Nonnon describes software

> allowing young pupils to control the motion of an electric train, and simultaneously to see its position graphed as a function of time, to enable them to acquire a graphical coding language to predict the interaction between the variables for distance, speed and time. The prototype software has been trialled, using pre-test and post-test to show a significant improvement in predicting and interpreting relationships between the three variables.

Garançon & Janvier report the entry stage into new research in **"the understanding of feedback systems with micro-computer software".** They formulate the general notion of a feed-back system as "a set of mathematically defined relations between variables" which can "generally be expressed as a set of differential or difference equations". They envisage the understanding of the system as a form of coordination of three representations of the system: an iconic representation of the feed-back loop relating the variables, the superimposition of the cartesian graphs of the variables as functions of time, and the phase plane diagram representing the implicit relationship between the variables. Current mathematical research into dynamical systems shows just how complex these systems can be and one looks forward with interest to the results of research into students' understanding of the specific systems designed for the research program.

2. Common links in the papers

It will already be apparent that the papers cover a wide range of activities. A closer inspection also shows that *no two papers cite a common reference*. (As a humorous aside, I found it pleasant to see that I am not the only author who refers to my own papers more than anyone else...) Despite the apparent anarchy that this may imply, there are certain underlying trends that can be seen.

2.1 Multiple linked representations

More than half the papers use software that links algebraic notation to a graphical representation, one links a real-world situation with a graph, one links the algebraic representation of an expression to its binary tree structure.

Kaput (1987) has suggested four sources of meaning in mathematics:

1. By transformations within, and operations on, a particular representational system,
2. By translation across mathematical representation systems,
3. By translation between mathematical and non-mathematical representations (such as natural language, visual images, etc.),
4. [Reflective abstraction] By the consolidation and reification of actions, procedures, and concepts into phenomenological objects

> which can then serve as the basis of new actions, procedures and concepts at a higher level.

It is helpful to review the papers within this framework to see their span over a range of activities. For instance, Nonnon links a graphical interpretation to the real world which

> "permet à l'élève d'acquérir un langage graphique de codage ... acquis au seul contact de la réalité, sans support verbal."

Boileau *et al* also wish to link the pupils' experience with mathematical concepts, this time through programming, whilst other papers concentrate more on translation *between* systems. When one of those systems is graphical, it is often seen as a more "intuitive" system. For example, Zehavi *et al.* comment that the main goal of their software is "creating intuitive readiness for future concepts".

Yet Goldenberg warns of difficulties with multi-representational software:

> "Common-sense supports the notion that the use of more than one representation of a function will help learners understand what remains less clear when only one representation is used. Presented thoughtfully, multiple linked representations increase redundancy and thus can reduce ambiguities that might be inherent in any single representation ... taken together, multiple representations should improve the fidelity of the whole message. The theoretical arguments ... arc reasonable enough, but they may not be valid."

His case questioning validity is based on his two subjects' misconceptions of the nature of graphs. Other research supports this concern. For example, Nachmias & Linn (1987) show that a computer-generated graphical representation of a cooling curve of liquid in real-time was misinterpreted by 30% of the children involved, because the large pixels on-screen gave the impression that the liquid remained at a constant temperature for a time and then suddenly dropped a little (to the next pixel level). These students believe in the absolute veracity of the computer. My own observations using computer graphs with older children students suggest that it is possible to discuss such limitations meaningfully, but there are clear indications of conceptual obstacles that need to be researched.

Lesh & Herre suggest that

> "Good problem-solvers are flexible in their use of various representational systems – they instinctively switch to the most efficient representation at any given point in the solution process".

Although preliminary empirical data shows the value of multiple linked representations, more data of how students of differing ability and experience cope will be of great value.

2.2 Microworlds and the Role of the Teacher

The vision of Papert was that, by giving children access to rich microworlds, such as programming in Logo, they would develop "powerful ideas". The reality of this vision is that they may not develop the powerful ideas that may be deemed desirable. For example, the children in the Sutherland study “did not naturally choose to use variables in their Logo programming” and teacher intervention was necessary to provoke suitable activities.

Dreyfus and Eisenberg comment on the “partial success” of several experiments using microworlds in “achieving a process of abstraction on the part of the student” and question “whether the framework needs to be revised”. They conclude that “this does not seem to be appropriate” as the studies were “rather short term” and “extremely high level activities are required for the processes involved in abstraction in general”. They hope that “longer and more systematic exposure to dual and triple representations of mathematical objects will achieve a clearer effect ... but at present this is simply speculation”.

2.3 The Notion of Variable

A noticeable feature of the papers is the variety of different meanings given to a variable. The pupils in Sutherland’s study all used (local) variables as inputs to procedures whilst those in Nelson’s used global variables with the command MAKE (which they subsequently forgot). Neither paper refers to the difference between a variable in algebra and in programming. (For instance, a Logo variable has a name **"X** and a value **:X**.)

Although Boileau *et al* consider elementary algebra as “minimalement l’algèbre des polynômes en une indeterminée, mais aussi les fonctions linéares, quadratiques, trigonometriques, exponentielles et logarithmiques”, they later speak of

> “des fonctions (toujours algorithmiquement calculables) definies sur des ensembles de nombres, éventuallement représentées par des tableaux de valeurs, des graphes cartésiens, ou des algorithmes de calcul”

which suggests the possibility of more general procedures. Interestingly, no paper mentions procedural functions even though, when “Al ... earns $6 per hour if he works 15 hours .... [and] gets paid time and a half for overtime” (in Lesh & Herre), his actual wage, for any)non-negative) number of hours, can be calculated in Logo as

```
TO WAGE "HOURS
IF HOURS<15 [OP 6 * :HOURS] [OP (6 * 15 ) + 9 * ( :HOURS – 15)]
END
```

or in structured BASIC as

```
DEF FNwage(x): IF x < 15 THEN := 6*x ELSE := 6*15+9*(x–15).
```

Either of these will easily generate a full table of values for his wage against the number of hours worked (normal and overtime), giving a more interesting and realistic function than the algebraic expression for overtime only.

In Thompson’s *Expressions Microworld*, letters have a more abstract use, standing either for numbers or other expressions, whilst, in some other papers, variables are parts of formulae related to graphical representations. Only Lesh &

Herre and Thompson & Thompson concern themselves with the *manipulation* of expressions. Lesh & Herre make the important observation that "the possibility of **first describing, then calculating** is one of the key features that distinguishes algebra from arithmetic". It is telling to note that Sutherland's Logo pupils without algebra experience use variables only to *store* numbers, not to manipulate them.

2.4 Research methodology

Of the eleven papers presented, only two have a traditional experimental v. control methodology, two use pre- and post-tests with the experimental students only whilst others used observational techniques or clinical interviews. Sutherland chose ethnographic methodology "as being the only one possible in an area where technology, pedagogy and the approach to mathematical content were all innovatory". Perhaps different techniques are required in different phases of research, with ethnographic methods more suited to the entry phase and a traditional methodology more suited to review, though this division is clearly not hard and fast.

3. Major questions that still need to be answered

3.1 Algebra in a computer environment

First and foremost we must begin to address ourselves to the role of algebra in a future computer-oriented paradigm. Most of the research presented here is concerned with the manner in which traditional algebra may be enhanced by the computer with little emphasis on a modern procedural approach. Many interesting functions such as the price of a postage stamp as a function of weight, are given procedurally rather than as a simple formula. Modern computer programs, such as the modelling program *Stella* (1986), allow functions to be typed in as formulae, as logical expressions, or even as piecewise straight graphs specified using an on-screen pointer under the control of a mouse. The new IHewlett Packard HP 28C symbolic calculator allows variables to have values including complex numbers, vectors, matrices and lists; thus a list of information such as the details required for drawing a graph (ranges, independent variable, number of points etc.) can be stored as a variable and recalled when required.

An important global question framing all our research should therefore be

> **How can we direct our use of the computer in mathematics education to the concentrate on the algebra of the future, in addition to the algebra of the past and present?**

In particular we should spend a little time thinking about the role of symbolic manipulators. My own hunch in using them is that they (at present) offer a powerful way of handling the *syntax*, but the user needs to have a coherent understanding of the *semantics.*

It is important also to address ourselves to the question of the needs of different user populations. Several of the research papers talk about pupils of "average ability" (a term which is sometimes a little difficult to interpret). Twenty years ago (in Britain at any rate) *pupils of average ability did not study algebra.* Leitzel & Demana (1987) suggest an *arithmetic* approach to algebra. May we sometimes be wasting our time looking at the difficulties of sections of the population for which formal algebra may be of no relevance? Should all children study the same kind of algebra, or do we need different types of algebra for different populations?

3.2 Multiple Linked Representations

Given the high profile of dynamically linked representations, it is clearly important to obtain far more empirical evidence of their use. In particular we should ask:

> **In what ways do students, of differing ages, abilities and experience, use dynamically linked representations in different curriculum contexts, and how do they conceptualize the relationships between the representations? What cognitive obstacles are likely to occur in their use?**
>
> **What is a suitable theory (or theories) underlying the provision of suitable developmental sequences?**
>
> **In what ways can multiple linked representations be integrated into the curriculum for learning, teaching, problem-solving, and assessment?**

Here we note that the links between representations can take differing forms, for example Garançon & Janvier view the understanding of feed-back systems as a coordination of three distinct representations, one of which is the *statement* of the problem (the feed-back loop) and others are *solutions.* Other systems simply *translate s*ymbolic information into graphical form.

> **For a given system, are there simple translations between two representations, or does the relationship involve some kind of solution process?**
>
> **Does the "understanding" of the relationship between two representations involve a direct logical relationship, or is it an intuitive one, or perhaps a combination of the two?**

It would be useful to debate the interplay between syntax and semantics, in terms of the classification proposed by Kaput, the notions of syntax and internal/external semantics of Boileau *et al* and the new evaluation of Dienes' principles as described by Lesh & Herre.

3.3 Programming

Two clearly distinct threads arise in the papers, one proposing specially designed software to enhance learning, the other to encourage constructive acts through programming. These may be seen as totally separate methods of approach, or as being *complementary,* fulfilling two different, but essential, roles. We ask:

> **In what way are programming and the use of prepared software complementary, and what constitutes an optimum combination of the two in terms of understanding and efficiency (time on task)?**

Boileau *et al* speak of a new language for learning algebra, whilst other papers use Logo. It is important to discuss what kind of computer language is appropriate, not just for doing algebra, but also for developing a growing awareness of algebraic structure during the learning process.

3.4 The Role of the Teacher

Lesh & Herre suggest that the use of certain software will encourage teachers to take a "mathematics laboratory" approach to learning and teaching, but Boileau *et al* remark that

> "En depit de ces progrès théoriques, les enseignants en mathématiques sont relativement dépourvus quand il s'agit d'aider les élèves à se représenter les relations des problèmes algébriques narratifs."

I suggest that *teachers are not convinced by theoretical research, but by ideas and materials that* ***work, for them,*** *in the classroom.* The role of the teacher should surely be an explicit part of our theories of mathematics education. With the complexity of the representational systems and the need for teachers to embrace computer technology, we must ask:

> **How can we encourage teachers to participate actively in our work so that our research is both relevant and suitable for implementation?**

3.5 Artificial Intelligence

Few of the papers mention the use of tutoring systems, though the *Expressions Microworld* and the symbol manipulator/function plotter *SAM* are both written in Lisp, which gives them the possibility of being used in a more diagnostic/predictive mode. The *Expressions Microworl*d has been explicitly written to *do nothing* if it is given an inappropriate command by the user, thus encouraging users to think about the consequences of their own actions. *SAM* can produce solution path "traces" to create many instructional capabilities and do other things that are intended to "help students go beyond *thinking* ***to think about thinking***". One view is that it is the teacher and the pupil who provide the intelligence, in a way that cannot be provided by the machine, another uses the machine to infer action from a database of knowledge.

Particularly in the case of algebra, which has both a syntactic and a semantic role to play in mathematics, we should ask:

> **In what ways can computer environments be designed and used to provide intelligent support to the learning process?**

3.6 Constructivism

This conference has constructivism as a major theme, and it is implicit in several of the articles, if not always explicit. My own belief is that learning is facilitated by the intelligent action of the pupil, with the teacher acting as a guide and mentor, and I have been struck by the power of the computer to provide a cybernetic environment that acts in a reasonable and predictive way to enable the pupil to build and test new concepts represented dynamically by the software. But do we all share this belief?

Davis (1986) poses the fundamental question:

> **Every educational use of computers is based upon someone's specific philosophy of what, exactly, is to be learned, and upon someone's philosophy of effective pedagogy. These "foundations" are, at present, extremely insecure.**
>
> **In the prescnt case, exactly how do we want our students to think about algebra?**

To this one must add:

> **llow can we use computers to encourage students' active participation to develop this algebraic thinking and to *think about thinking* ?**

4. The Way Ahead

I am aware that although some of the questions I have highlighted are phrased as research questions, others are not. Our discussion must include an attempt to focus on specific research hypotheses. It was part of my brief to seek indications from the papers as to how to tackle the highlighted problems. As most of the authors concentrate on putting over their own message in a limited seven page span, it would not be fair to expect the papers to be addressed explicitly to queslions formulated after the papers were written, however, I am confident that the collective wisdom and experience of the authors may be brought to bear in the discussion at P.M.E.

References

Davis R., 1986: What "Algebra" Should Students Learn?, *Journal of Mathematical Behaviour*, 5, 21-24.

Kaput, J.J., 1987: Draft paper for Research Agenda Project, Athens, Georgia.
Leitzel, J.R. & Demana, F., 1988: Establishing fundamental concepts in algebra through numerical problem solving. *NCTM 1988 Yearbook (to appear).*
Mason, Burton & Stacey, 1982: *Thinking Mathematically,* Addison-Wesley.
Nachmias, R. & Linn, M.C., 1987: Evaluations of Science Laboratory Data: The Role of Computer-Presented Information, *Journal of Research in Science Teaching (to appear).*
Stella, 1986: High Perforrnance Systems, 13 Dartmouth College Highway, Lyme NH 03768.