Using the computer as an environment for building and testing mathematical concepts: A Tribute to Richard Skemp

David Tall

Mathematics Education Research Centre
University of Warwick
England

Introduction

In this paper I shall show how the computer may be used to provide an environment for building and testing mathematical concepts as part of a long-term learning schema following the theory of Skemp, (1962, 1971, 1976, 1979). I do this in tribute to the inspiration I have received from working with Richard Skemp and reading his publications over the major part of my time in mathematics education. History may record that it was my fortune to be his last Ph.D. student before his retirement as Professor of Educational Theory at Warwick University. It is an honour that I cherish, and a responsibility that will be hard to maintain at a fraction of the standards that he has set in scholarship and insight.

Background

Fifteen years ago I was to give my first course of lectures on “The development of mathematical concepts” to undergraduates who were mathematics majors with excellent records in mathematics but no experience in studying how people learn. The course was based on the difference between the formal, the historical, and the cognitive development of mathematical concepts. The formal mathematics was familiar to students, the historical and cultural side could be taken from a number of sensible mathematical histories, but the cognitive development caused some difficulties. In the early seventies there were five major thinkers in the psychology of education who gave highly relevant insights into mathematics education : Piaget, Dienes, Bruner, Ausubel and Skemp. For my undergraduates, with little knowledge of children, Piaget proved difficult to penetrate. Although we could talk about assimilation and accommodation of concepts in a meaningful way, the notion of the stages: sensori-motor, concrete-operational and formal-operational, was purely a theoretical construct. To these students a “concrete operational child” was one who could conserve number, quantity, volume, etc, and could argue logically, provided there were concrete materials available to represent the concepts physically. The students had an enjoyable time playing with Dienes’ logic blocks and algebraic experience materials for factoring quadratics. Yet the precise nature of how this concrete

Published in *Papers in Honour of Richard Skemp*, 21–36, Warwick (1986).

experience is translated into formal ideas remained something of an enigma. We considered some of Bruner's papers on a theory of instruction and the notion of discovery and we made reference to the text on educational psychology by Ausubel. But the textbook that made the course meaningful to the students was a popular little paperback that could be bought in W. H. Smith's bookshop by the average browser: Richard Skemp's *Psychology of Learning Mathematics.* The passage of time has seen this book distilled and revised, translated into other languages, and now brought out in a new American edition. Fifteen years after its first publication it remains a book to stimulate thought and discussion about the way we learn mathematics. Its major value for so many people is that it is not about mathematical learning at specific levels of development, but it has implications at all levels. Later it was to be followed by *Intelligence, Learning and Action*, again a profound book that has wide implications.

The difference between Skemp's approach and others is highlighted by his desire to make his work understandable not only to theorists, but also to educational practitioners. Ausubel (1968), for example, desires to give a clarity to his theories by using technical terms that are endowed with special definitions. He describes concepts as "objects, events, situations or properties that possess common criterial attributes (despite diversity along other dimensions or attributes) and are designated by some sign or symbol, typically a word with generic meaning". Skemp (1971), on the other hand, believes strongly in using evocative terminology and giving it a special meaning or "interiority" in the special context. He remarks that, "though the term 'concept' is widely used, it is not easy to define, nor... is a direct definition the best way to convey its meaning." He goes on to elucidate two principles:

1) Concepts of a higher order than those which a person already has cannot be communicated to him by a definition, only by arranging for him to encounter a suitable number of examples.

2) Since in mathematics these examples are almost invariably other concepts, it must first be ensured that these are already formed in the mind of the learner.

These principles taxed the minds of myself and some of my university mathematics colleagues for some time in the early seventies. I well remember a colleague teaching a group theory course by not mentioning the term 'group' at the outset, and beginning with various examples that would eventually form the concept later: permutations of finite sets, geometrical transformations, and so on. The trouble was that, without some knowledge of where they were heading, the students seemed to be confused by the diversity of the examples in the early stages. Clearly an organising principle was needed that would help in the abstraction of higher order concepts.

Ausubel *et al* (1968) had already proposed such a principle: the *advance organiser*, which is "introductory material presented in advance of, and at a higher level of generality, inclusiveness, and abstraction than the learning task itself, and explicitly related both to existing relevant ideas in cognitive structure and to the learning task.” Another colleague at university considered that the ideas in the linear algebra course were essentially very simple, and began his course with a preview lecture that covered all the essential principles in one go. He found that the students were somewhat perplexed by the lecture. At this stage they seemed to lack the relevant higher order ideas to comprehend the simplicity of the structure.

Advance organisers, by definition, require the student to possess knowledge appropriate to the new task, which is at a higher level of generality than the task itself. For example, Ausubel applied his theory to the learning of a second language where the learner may already have higher level concepts available from the study of the first language: sentence construction, parts of speech, tense of verbs, and so on. In mathematics there may occur times when appropriate higher level concepts are not available to the students.

In such cases, how are we to introduce the learner to the new concepts? Dienes (1960) suggests that physical manipulation of his materials at the concrete operational stage may provide experiences from which the higher level concepts may be abstracted at a later stage. He suggests a number of principles on which to build up ideas from concrete experience to formal abstraction. But this is not the only possible strategy.

In Dienes and Jeeves (1965) a contrary principle is put forward, suggesting that, in some cases, more effective learning may be achieved by first introducing the learner to a structure that is more general and more complex. By being thrown in, at the ‘deep-end’, the authors suggest that the learner is more likely to use a higher order ‘operational structuring’ approach to cope with a problem.

Thus Dienes is suggesting two different kinds of approach, one which builds up in a careful and structured way from concrete materials, the other that sets the learner in a problem area to discover the higher level structure from his or her investigations. Again, how are we to reconcile these two different pieces of advice, and how are we to proceed best when we introduce students to new areas of mathematics?

In *Intelligence, learning and action*, Richard Skemp suggests a clarification of the concrete/formal dichotomy in his modes of building and testing (figure 1). He remarks that, as currently taught, pure mathematics relies heavily on mode (iii), in varying degrees on mode (ii) and not at all on mode (i). However, if one interprets the idea of initial concrete activities being centred in mode (i), aided by explanation and discussion in mode (ii), then the computer gives us a totally new way of attacking the learning of mathematical concepts.

The important first phase in learning new concepts is not necessarily through manipulating concepts that are represented *physically* in a concrete sense, but *externally,* in a manner which allows predictable manipulations for building and

REALITY BUILDING	REALITY TESTING
Mode (i) from our own encounters with actuality: *experience*	***Mode (i)*** against expectation of events in actuality: *experiment*
Mode (ii) from the realities of others: *communication*	*Mode (ii)* comparison with the realities of others: *discussion.*
Mode (iii) from within, by formation of higher order concepts. by extrapolation, imagination, intuition: *creativity*	***Mode (iii)*** comparison with one's own existing knowledge and beliefs: *internal consistency*.

Figure 1

testing. This predictable environment for building and testing can be provided by suitable software on a computer.

The computer has capabilities that complement those of the human mind. Where the human mind is prone to error, yet able to make leaps of insight, the computer is consistent (provided it is properly programmed) and predictable. It is this very consistency and predictability that make it, in Skemp's terminology, an *environment for building and testing mathematical concepts*. For, if the computer is programmed to carry out mathematical processes in a way which makes them transparent to the user, then the programmes can be used in mode (i), with explanations in mode (ii), to gain experiences of concepts in action, forming a cognitive base for the later building and testing of the formalities of the theory in modes (ii) and (iii). One may *build* concepts by considering a number of examples (and non-examples) of a process in action to observe regularities and abstract the underlying generalities, and in a complementary fashion, one may *test* concepts by conjecturing what may happen in as yet untried situations before carrying them out to check the conjectures.

Generic organisers

An environment that provides the user the facilities of manipulating examples of a process or concept I term a *generic organiser*. The term "generic" means that the learner's attention is directed at certain aspects of the examples which embody the more abstract concept. Thus the equality 3+2=2+3 may be seen as a *specific* example of arithmetic in which two additions give the same result, or as a *generic* example of the commutative property of addition. The generic example is seen as a representative of a whole class of examples which embody the general property.

Computer programs can show not only examples of concepts, but also, through dynamic action, they can show examples of mathematical *processes*. In some instances the processes become encapsulated as mathematical concepts. Thus my program GRADIENT (Tall 1986a) which shows a secant clicking along a curve, simultaneously plotting the gradient of the secant as a point, is carrying out a mathematical *process*. But the finished product is a static graph representing the gradient function of the curve, and this is now a mathematical *concept*. Thus computer programs can provide an enviroment that enables teachers to demonstrate mathematical concepts and processes, and students to explore them. During the demonstrations and explorations it is possible to focus not only on the particular example being constructed by the computer at the time, but also to see it as being a typical representative of a class of examples. By seeing a wide variety, not only of examples, but of non-examples, using the speed and calculation power of the computer at the behest of the teacher and pupil, it is possible to focus on the more general concept that is represented by the whole class of examples and get a true feeling for the concept that can form the basis for more formal later work.

Before the advent of the computer, a common philosophy of teaching was based on the notion that the French call the “didactic triangle” between the pupil, the teacher and the mathematics (figure 2):

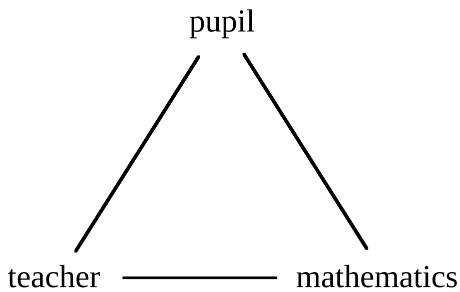

Figure 2 : The Didactic Triangle

The mathematics is part of a shared knowledge system, shared by those who have already learnt to understand it. The respresentative of this culture in the classroom is the teacher. The mathematics is *in the mind* of the teacher and the only externalized physical represention are usually in a text book. Here the mathematics is *static* in fixed words and pictures. The only *dynamic* representation is through the verbal explanation of the teacher and any diagrams that may be drawn.

The Didactic Tetrahedron

The introduction of the computer brings a new dimension into the learning situation. There are now four major components, which may be viewed as forming a tetrahedron in a suitable educational context (figure 3):

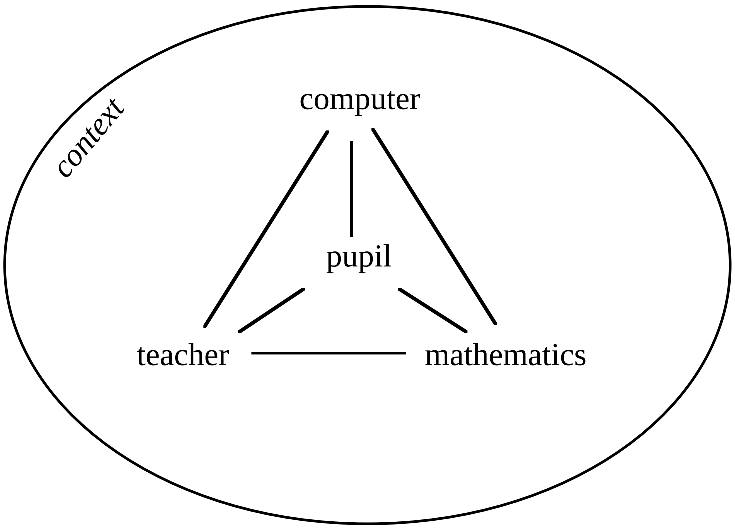

Figure 3 : The Didactic Tetrahedron

It is assumed that the computer has appropriate software available to represent the mathematics, and that this software is designed in a manner that makes the mathematics as *explicit* as possible. It must show the *processes* of the mathematics as well as giving the final results of any calculation.

If the computer software is in the form of a generic organiser, then it may be used in a flexible way. I have seen my own software given to pupils to solve problems without any explanation as to how it should be used. As a challenge, in the right context, this ‘deep-end’ approach can be most effective. My own preference is for an initial element of *teaching* and *discussion* with the teacher using the organiser to demonstrate examples, slowing down the computer action to explain what is happening, and pausing on occasion in the middle of a routine when an interesting point is reached that is worthy of further discussion. (For this reason I design computer programs with speed of action always under the control of the user.) The intention of discussion at this stage is a *negotiation of meaning*. The idea is to help the students form their own concept images in a way that is likely to agree with the interpretation of the mathematical community. This may be done through a Socratic dialogue between teacher and pupils which is enhanced by the addition of the computer. The mathematics is no longer just in the head of the teacher, or statically recorded in a book. It has an external representation on the computer as a dynamic process, under the control of the user, who may be the teacher, the pupil, or a combination of people working together. Concepts may be *built* by seeing examples in action, and *tested* by predicting what will happen on artfully chosen examples before letting the computer carry them out.

The *enhanced Socratic mode* of teaching that I use with my own generic organisers begins with teacher demonstration of the concepts on the computer and dialogue between teacher and pupils in a context that encourages enquiry and cooperation. At this stage only one computer is required for the whole class (though younger children are best helped in small groups). At the earliest opportunity, individual pupils will be typing in their own suggestions which arise as a result of the dialogue and, as they gain in confidence, the teacher plays a less central role. There may come a phase of operation in which the pupils are using the generic organiser for their own investigations. In larger classes this may require more computers to allow all the students access, though it is often possible to organise a circuit system enabling pupils to take it in turns to work together in small groups. At this stage the operative part of the didactic tetrahedron is the relationship between the pupil and the mathematical ideas, as represented on the computer by the generic examples. The teacher takes no directive role, being available only to answer questions which may arise in the course of the student investigations. At a later, review stage, further discussion with the pupils is sensible, to probe their ideas and make certain that their concept image is appropriate for the wider mathematical community. In this way the students come to terms with the ideas through experience and build their concepts in a way which is likely to be potentially meaningful.

Examples of generic organisers

Fractions

In *Tall 1986b* a generic organiser is described that was written to encourage discussion and exploration of the notion of a fraction. It is my belief that the child's understanding may be hampered by the limitations of its own physical ability. For example, it may be possible for the child to visualize a cake cut into seven equal pieces, but not to carry out the accurate physical subdivision. The vernacular use of fraction names in uses such as "give me the bigger half" is then emphasised by actual experience where fractions tend to be cut inaccurately. The computer is not necessarily helpful here. A computer picture is made up of individual pixels, so if a line length 30 pixels is to be cut into four equal parts, some will be seven pixels long, and some eight. The computer is hardly more accurate than the child! However, if a circle is cut up into sectors, although it is again performed inaccurately on the computer screen, it is possible to maintain an illusion that it is being done accurately, even if the picture is not exact. Using this idea I wrote a simple program for subdividing a "cake" into a specified number of equal pieces and allowing the user to take as many pieces as are required. The cutting process is counted out, and the number of pieces required are counted on-screen, using the usual symbols for fractions. Children have a great sense of humour. They may cut a cake into five equal pieces and then demand to have eleven of them! The child is not restricted to demanding less than a whole cake, and so can gain direct experience of fractions greater than one. The computer complains "I'll have to cut more cake" and stoically draws two more cakes, cutting each into five equal parts and counting out the required number of pieces. In figure 4 the computer has counted the five pieces on the first cake, has moved on to the second and counted piece six and is now on seven. The numerator of the fraction changes as the sector is counted and the process will be completed after counting eleven pieces by the remaining pieces on the last cake being eaten away.

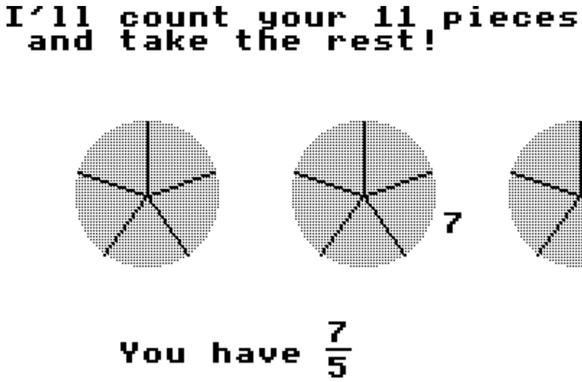

Figure 4

The static picture in figure 4 is not very informative. It only comes to life when seen dynamically on the computer screen. By itself the program is quite limited, but as a focus for discussion, exploration and explanation it has proved valuable in giving an enriched concept image of the notion of fraction, including fractions greater than one. Routines are also included in the program to show the idea of equivalent fractions by cutting up the cake in equivalent ways.

Algebraic notation

A generic organiser developed for algebraic notation in conjunction with Michael Thomas *(Tall & Thomas 1986)* shows another aspect of introducing a mathematical concept by mode (i) exploration. Here we wished to give a mental image of the notion of a variable as a labelled box into which a value of the variable could be placed. The children, aged 11 or 12, performed a variety of related tasks, some using a physical representation of a 'maths machine' in the form of a card with rectangular boxes marked on it, which could be used as labelled stores for numbers. They also wrote short BASIC programs whose purpose was to give numerical values to variables such as X,Y and to calculate numerical values of expressions such as X+Y, 2*X+Y or 2*(X+Y). The writing of programs is a very definite way in which the computer can be used to build and test the concept of a variable using the predictability of the computer language. However, note that BASIC is not a language that allows actual algebraic manipulations of the symbols, so it can *only* act in a generic way, using algebraic notation in the program to represent specific numbers at any given time.

In addition to the cardboard 'maths machine' and the programming activities, the pupils also used a computer program which represented the variable stores, labels and contents on screen. Some stores were allowed varied inputs and represented 'variables', other stores had fixed contents, representing 'constants' and others could be labelled with algebraic expressions, giving algebraic functions which were automatically calculated by the program. (Figure 5.)

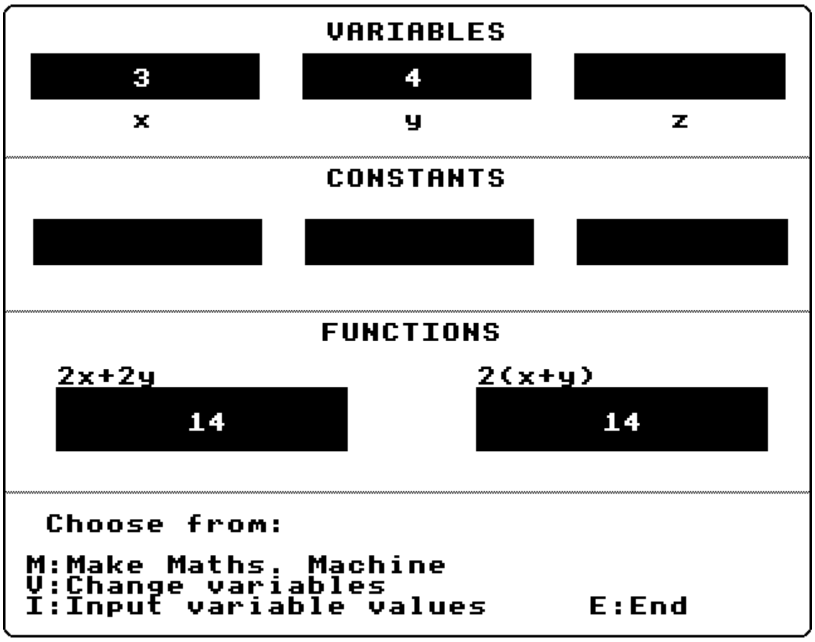

Figure 5 : The Algebraic Maths Machine

The expressions here could be ordinary algebra, allowing implicit multiplication, such as 2x, as well as explicit BASIC notation and the pupils could use the program to investigate problems experimentally, for instance, "when is the value of 2*x*+3 bigger than 10?".

The computer 'Maths Machine' is a generic organiser in the sense that it enables the pupil to picture the mathematical concept of a variable. (It is less effective in showing, for example, how the calculations are actually carried out, but this is done by the activities with the cardboard maths machine and the programming.)

After a three week module of work using the above materials, the children were performing at a significantly higher level than the average performance of pupils several years older in certain tasks requiring an understanding of algebraic notation. (See *Thomas, 1985*, for further details.)

Graphic Calculus

The major area where generic organisers have been developed to aid in the formation of higher order concepts is in the learning of the calculus. Following Skemp (1962), in one of my first publications in mathematical education I attempted to develop a long-term learning schema for the calculus (Tall 1975). This was based on an attempt to reformulate the calculus on more meaningful terms, based on a theoretical analysis of the concepts. However, research in Schwarzenberger and Tall (1978) revealed genuine cognitive difficulties in the subject. These were further reported in Tall & Vinner (1981) and similar phenomena were noted in Cornu (1981, 1983). Students have particular difficulties with the limiting concept and with the interpretation of words whose everyday meaning is different from the technical mathematical meaning.

This problem is made worse by using a formal approach to mathematics at a stage when the students lack the sophistication to be able to cope with it. Even so-called "intuitive" approaches to the subject are often based on a formal order of development. For example, the gradient of a general graph is formally defined in terms of the derivative, which is itself defined formally as a limit. Most develoments of the calculus therefore precede the notion of the gradient of a graph by a discussion of the limiting process which the students may fail to understand. Instead they form their own *concept image* in an idiosyncratic manner.

The generic organisers for the calculus are designed to present the learner with experiences that enable them to develop suitable mental images of the mathematical concepts. Instead of a *formal approach*, in which the mathematical pre-requisites are presented in a non-meaningful context before the main concepts are described, they are part of a *cognitive approach* in which a global gestalt of the main concept is given in the early stages. For example, instead of beginning the theory of differentiation with a discussion on limits, one may present a global gestalt of the notion of the gradient of the graph using the generic organiser GRADIENT to visualize the changing gradient as a secant steps along the graph. It is even better if the student already has cognitive experiences of magnifying the graph and sees the graph as being "locally straight".

A cognitive approach to the calculus

The approach advocated in GRAPHIC CALCULUS has been fully documented elsewhere, in a series of six articles in *Mathematics Teaching* beginning with Tall (1985), and in the program documentation itself, Tall (1986a). For the purposes of this article, which is mainly concerned with the theory of building and testing of concepts using generic organisers, I shall concentrate on two aspects, the concept image of the gradient of a given graph and the converse problem, to find the graph, given the gradient.

The first stage is to establish the idea of the gradient of a curved graph. The program MAGNIFY allows the user to first draw a graph and then select any part of the graph and magnify it. Figure 6 shows the graph of $f(x)=x^2$ drawn from $x=-2$ to 2, and a small part of the curve centred on $x=1$ magnified. The magnified portion looks (almost) straight and has gradient 2.

If students are left to magnify graphs of their choice, they will soon discover that almost all functions they are able to type into the computer have the property that small parts of their graphs magnify to look nearly straight. However, without intervention from the teacher they may believe this to be a general property of graphs and form an inadequate concept image. They may never have drawn graphs that do not have this property other than, perhaps,

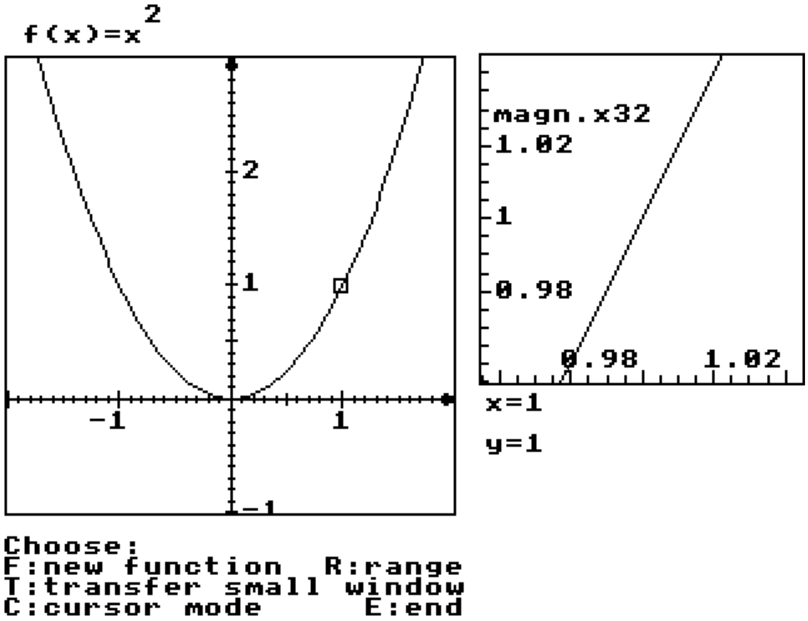

Figure 6: Magnifying a locally straight graph

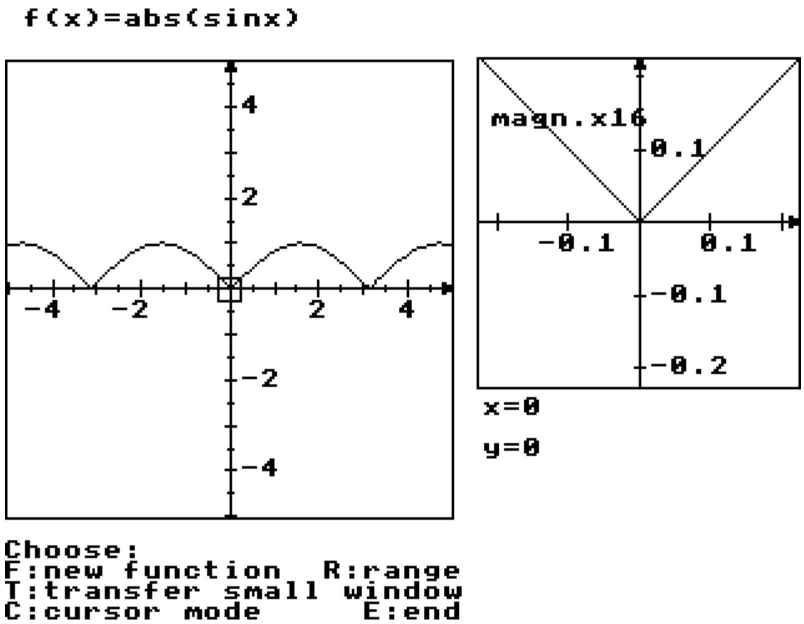

Figure 7: graph with corners

$f(x)=|x|$ (which must be typed into the computer as abs(x)). But the single example of the absolute value function can be combined with others to give more interesting graphs, such as $f(x)$=abs(sin$x$), which magnifies at the origin to show two different gradients to the left and right. (Figure 7).

This now liberates the possibilities of visual imagery. Consider the graph

y=sin$x$+abs(sin(100x))/100.

It consists of the graph y=sin$x$, with tiny oscillations built on from abs(sin(100x))/100. To a standard scale the graph looks not much different on

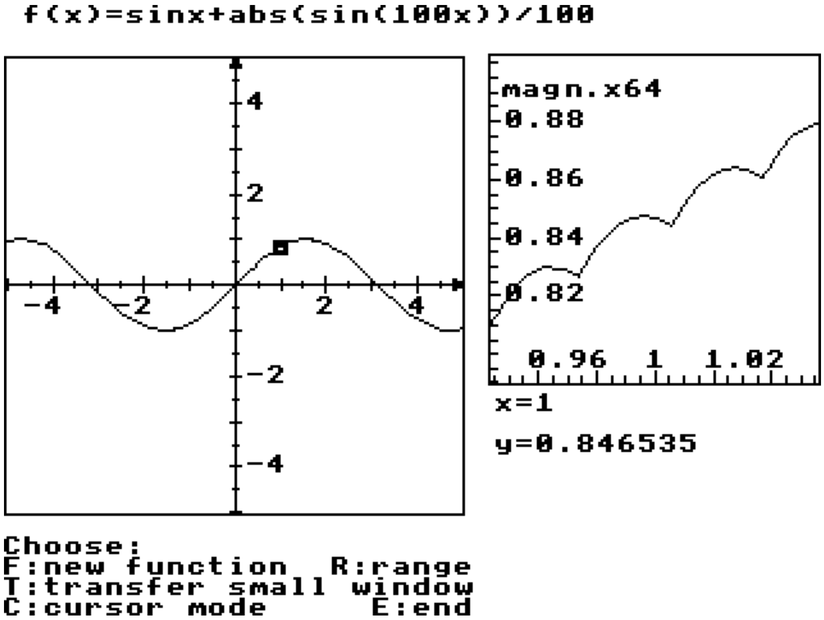

Figure 8: A graph which magnifies to reveal corners

the computer screen from y=sin$x$, but a suitable magnification shows corners at regular intervals (figure 8).

By seeing examples such as these, the student is assisted in developing a concept image of the types of graph that are discussed in the calculus. There are those which look “locally straight” and there are others, which may look not much different to the naked eye, yet which have corners. It is quite possible to discuss more complicated examples fairly early on in the calculus, though this is very much a matter of taste to be decided by the teacher. What is certain is that these more interesting examples need a great deal of sophistication to invent and may not arise in free explorations by the students without guidance from the teacher. As Bruner stated in *The Relevance of Education* (1974),

> It seems to me highly unlikely... that one would expect each organism tto rediscover the totality of its culture...

Even given the powerful visualisations available in a generic organiser such as MAGNIFY or GRADIENT, these are only made apparent by the sensitive guidance of a gifted teacher.

After experiences with MAGNIFY, the program GRADIENT allows students to see the gradient of a “locally straight” curve built up dynamically as a secant through nearby points (x, $f(x)$), ($x+c$, $f(x+c)$) clicks along the curve for variable x and fixed c. As each secant is drawn, its gradient is plotted as a point, leaving behind a trace of points that outlines the gradient graph (figure 9).

By experiences such as these, pupils learning the calculus have an increased chance of linking the formal manipulations of the calculus mentally with a geometric visualization of the process. For example, in the thesis Tall (1986c), it is shown that students using the gradient program are better able to sketch the

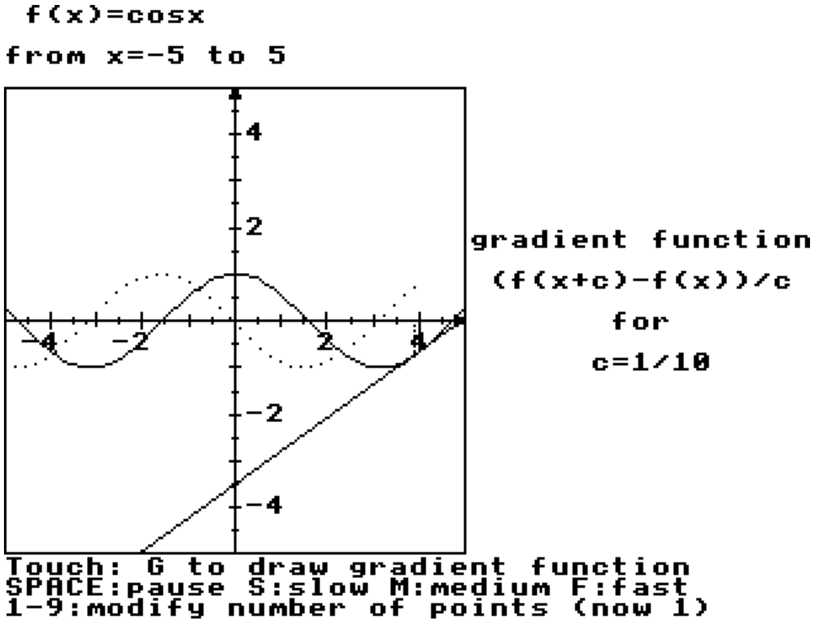

Figure 9: Sketching the gradient of cos*x*

gradient of a given graph than those who have had a more traditional introduction to the theory. They are also better able to carry out the reverse process of recognising the original function given the gradient. The striking statistic in this study is the number of students in the exercise succeeding in both tasks (which was set at obtaining over 75% of the marks for sketching the gradients of four graphs and correctly identifying the original function given the gradient graph). There were 26 out of 42 experimental students (62%) who were successful in both, but only 2 out of 72 control students (3%).

The students' success was dependent on the suitable use of the generic organisers. They worked well in two schools following a prescribed course of action designed to develop the concept imagery of the gradient of a graph. In a third school, where the program was not used for demonstration and the students were given only a single opportunity to draw a gradient, the results were significantly worse than the control students who did not use the computer.

The need for an organising agent

As we have seen, generic organisers by themselves do not guarantee abstraction of the general concept embodied in the organiser. Left to their own devices, students may either miss the point of the organiser or, more seriously, the organiser may be misused and non-generic noise embodied in the implementation may distract the user and lead to cognitive obstacles. The human mind is a very powerful pattern-detecting mechanism and may easily alight on an underlying regularity that is not intended for abstraction. For example, the functions that are typed into the GRAPHIC CALCULUS programs are all combinations of standard functions and, with the exception of the absolute value ABS, the integer part INT and the signum function SGN, all these tend to be continuous and differentiable. My experience is that students do not draw examples of graphs with 'corners' if they are left to their own devices. Thus exploration without guidance could lead to either to the belief that every function is differentiable, or that every function is differentiable except at a number of isolated exceptional points. Indeed, the latter was the commonly accepted belief amongst professional mathematicians in previous centuries.

A generic organiser is therefore only *potentially* meaningful, in the sense of Ausubel. The learner usually requires an external *organising agent* in the shape of guidance from a teacher, textbook, or some other agency to point to the salient generic features and away from misleading factors. In the preceding discussion I have referred to the enhanced Socratic mode in which the teacher plays the role of the organising agent. It may happen that the expert knowledge systems of the Fifth generation of computing may provide facilities which can act the part of the organising agent. Languages like PROLOG promise to move in this direction but have yet to fulfil their promise. At the moment there is a clear divide between the modes of thinking of man and machine. The brain of man acts both globally and sequentially. Some neurologists describe the global action of the right hemisphere and the sequential action of the right. Even though this is open to some dispute, it is a useful metaphor. The role of the organising agent demands a global facility and, at the moment, the computer seems to offer only the complementary power of a very fast metaphorical left brain. Nevertheless one should look forward to the time when computer systems will include guiding and monitoring facilities in learning, with generic organisers available as "desk-top" options, to be selected and used for learning in the manner that the calculator is available in the classroom today.

A long-term learning schema using generic organisers

And so I return to the original topic of my first research into mathematics education: the design of a long-term learning schema for the calculus, as an exemplar of long-term learning schemas leading to more advanced formal mathematical concepts. I have laid out the framework of the plan in Tall (1985) *et seq.* Generic organisers play a vital role in moving from the specific example, to the generic example, representing a typical element of the whole class of examples, to the general concept which encapsulates the whole class as a single higher level entity. It is this movement to a higher level of conceptual thinking that creates the greatest difficulty in mathematics. Once the move has been made, the mathematics becomes greatly simplified. It is this fundamental divide that separates out those that can, from those that cannot do mathematics. Those that 'can' see the simplicity of the general abstract idea, those that 'cannot' see the plethora of individual examples, each endowed with the noise of the particular case that makes it difficult to obtain any sense from such complex diversity.

Richard Skemp has taught me, above all, that it is not the complexification of ideas that marks progress, it is the distillation and simplification of ideas, making them available to a wider class of users, guided by the simple abstraction of the unifying ideal. Compare the theories of motion of the Greeks, with the movement of the heavens described by the “harmony of the spheres”, in which circular motion is superimposed on other circular motion, with the simplicity of Newton’s laws. The observation that acceleration is proportional to the force is a simple one, but it is also profound.

In this way we must look for unifying ideas in the calculus. These unifying ideas are not found in practising the use of a long list of formulae for differentiation, or the greater number of techniques for integration, but in the knowledge that the calculus is about the rate of change (differentiation), cumulative growth (integration) and the inverse relationship between them (the fundamental theorem). Cognitive support for all three ideas, and for many more besides, can be given by the generic organisers of GRAPHIC CALCULUS. Each organiser is intended to offer an immediate intuitive grasp of the idea, embodied initially in a single example, then refined and developed in further examples, in a manner that can readily be apprehended by the learner. Of necessity, this operates in a simplified context, appropriate for the learner at his current state of cognitive development. It therefore may tell the truth, but not the whole truth. But a good generic organiser can provide not only the intuitive grasp of concepts for the present, it can sow the seeds for the later development of the theory.

For example, the organisers MAGNIFY and GRADIENT together show the idea that a differentiable function is one that looks “locally straight”. This leads naturally into the concept of a local linear approximation and (years later) to the idea that differentiable manifolds are “locally flat”. Likewise, the blancmange function (a specific example of a function which is *nowhere* locally straight) leads on to a natural formal proof of the existence of an everywhere continuous, nowhere differentiable function (Tall 1982).

This example exhibits a feature of the use of generic organisers in long-term learning schema. The organiser GRADIENT works on the principle that the gradient of a curve can be visualized by looking along the graph and plotting the gradient of a moving secant. By adding a tiny multiple of the blancmange function b(x) to any differentiable function $f(x)$, one gets a function $f(x)+kb(x)$ (where k is very small), which is non-differentiable. The graphs of the functions $f(x)$ and $f(x)+kb(x)$ look indistinguishable drawn to a standard scale on a computer screen. Yet one is differentiable everywhere and one differentiable nowhere. Thus the initial idea of looking along a curve to visualize its gradient is theoretically unsatisfactory. Two graphs can look alike in a given picture yet one has a derivative and one does not. The organisational system contains the seeds of the eventual replacement of the generic organiser. It leads to a higher plane where one realizes the need for a more rigorous theoretical formulation.

Thus an organiser in a long-term learning schema acts in a manner which has a Piagetian stage structure. First it must be used in an environment where a simple formulation is possible, giving a sense of equilibrium to the learner. Later a dissonant property may be encountered which causes conflict and requires mental reconstruction to move into a new and richer level of equilibrium. The function of a good long-term generic organiser is first to be directly relevant to the current cognitive state of the learner, yet to contain the seeds of more subtle ideas that lead into later formal theory when, and if, that proves necessary.

The full long-term learning schema using generic organisers to move up to higher order concepts is built on *cognitive*, not *logical*, principles, as enunciated by Richard Skemp in 1971:

> Some reformers try to present mathematics as a logical development. This approach is laudable in that it aims to show that mathematics is sensible and not arbitrary, but it is mistaken in two ways. First it confuses the logical and the psychological approaches. The main purpose of a logical approach is to convince doubters; that of a psychological one is to bring about understanding. Second, it gives only the end-product of mathematical discovery ('this is it, all you have to do is learn it'), and fails to bring about in the learner those processes by which mathematical discoveries are made. It teaches mathematical thought, not mathematical thinking.

It is to be hoped that the appropriate use of generic organisers will provide a rich environment to stimulate mathematical thinking and build a fuller understanding.

References

Ausubel, Novak & Hanesian 1968 : *Educational Psychology : A Cognitive View*. Holt, Rinehart & Winston.

Bruner J.S. 1974 : *The Relevance of Education*, Penguin.

Dienes Z.P. 1960 : *Building Up Mathematics*, Hutchinson.

Deines Z.P. & Jeeves M.A. 1965 : *Thinking in Structures*, Hutchinson.

Cornu B. 1981 : 'Apprentisage de la notion de limite: modèles spontanées et modèles propres', *Actes du Cinquième Colloque du Groupe Internationale PME*, Grenoble 322-326.

Cornu B. 1983 : *Apprentisage de la notion de limite : Conceptions et Obstacles*, Thèse de Doctorate, Grenoble.

Schwarzenberger R.L.E. & Tall D.O. 1978 : 'Conflict in the learning of real numbers and limits', *Mathematics Teaching*, 82 44-49.

Skemp R.R. 1962 : 'The need for a schematic learning theory', *Brit. J. Educ. Psych.*, xxxii, 133-142.

Skemp R.R. 1971 : *The Psychology of Learning Mathematics*, Penguin Books Ltd.

Skemp R.R. 1976: 'Relational understanding and instrumental understanding', *Mathematics Teaching* 77.

Skemp R.R. 1979: *Intelligence, Learning, and Action*, Wiley.

Tall D.O. 1975 : 'A long-term learning schema for calculus/analysis', *Mathematical Education forTeaching*, 2,2 3-16.

Tall D.O. 1982 : 'The blancmange function, continuous everywhere, but differentiable nowhere',*Mathematical Gazette*, 66, 11-22.

Tall D.O. 1985 : 'Understanding the Calculus', *Mathematics Teaching*, 110 49-53.

Tall D.O. 1986a : *Graphic Calculus I, II & III,* Glentop Publishing.
Tall D.O. 1986b : 'Using the computer to foster understanding of mathematical concepts: talking about fractions', *Micromath* 2,2.
Tall D.O. 1986c : *Building and Testing a Cognitive Approach to the Calculus using Interactive Computer Graphics,* Ph.D. thesis, University of Warwick.
Tall D.O. & Thomas M.O.J. 1986 : 'Playing algebra with the computer', *Exploring Mathematics with Microcomputers* (ed. Bufton N.), 59-74, Council for Educational Technology.
Tall D.O. & Vinner S. 1981: 'Concept image and concept definition in mathematics, with particular reference to limits and continuity', *Educational Studies in Mathematics* 12 151-169.
Thomas M.O.J. 1985 : The Effects of BASIC Computer Programming on the Understanding of the Use of Letters as Variables in Algebra. M.Sc. thesis, Warwick.