Advanced Mathematical Thinking
& The Computer

David Tall

Mathematics Education Research Centre
University of Warwick
COVENTRY CV4 7AL
e-mail:seral@csv.warwick.ac.uk

Introduction

This presentation considers the developing role of the computer in advanced mathematical thinking, focusing on the learning of advanced mathematics by university students. It will consider different characteristics of advanced mathematical thinking and report research into the use of the computer in undergraduate learning whilst being cognisant of the differing needs and abilities of students concerned.

Two Types of Advanced Mathematical Thinking

In some ways university mathematics is an *extension* of the methods of school mathematics, particularly mathematical *techniques* which involve the use of numeric and algebraic methods (although the latter are less well covered in more recent years in school than they may have been in earlier times). In other ways university mathematics requires a *reconstruction* of old ideas to take into account *formal* methods of building theories from definitions and proof. A significant problem here is that mathematicians – who have already developed formal techniques and see the power of the method – have a very different view of mathematics from students who must build towards this more subtle form of thinking from their current developmental position.

The difference between these two forms of advanced mathematical thinking can be instanced by the case of linear algebra (figure 1). Here it is assumed that the student has already met various numeric, symbolic and graphic approaches to linear equations in one and two variables. To deal with the technicalities of solving m linear equations in n unknowns can be carried out by extending familiar techniques of manipulating equations. There are certain difficulties, one being the students' perceived difficulties of extending visualisations in R^2 and R^3 to higher dimensions. However, the "usual rules of manipulating linear equations" naturally extend to the general case in higher dimensions. Alternatively, the theory may be built formally using the definition of a (finite dimensional) vector space V over a field F. The axioms for a vector space (abstracted by previous generations of mathematicians) are
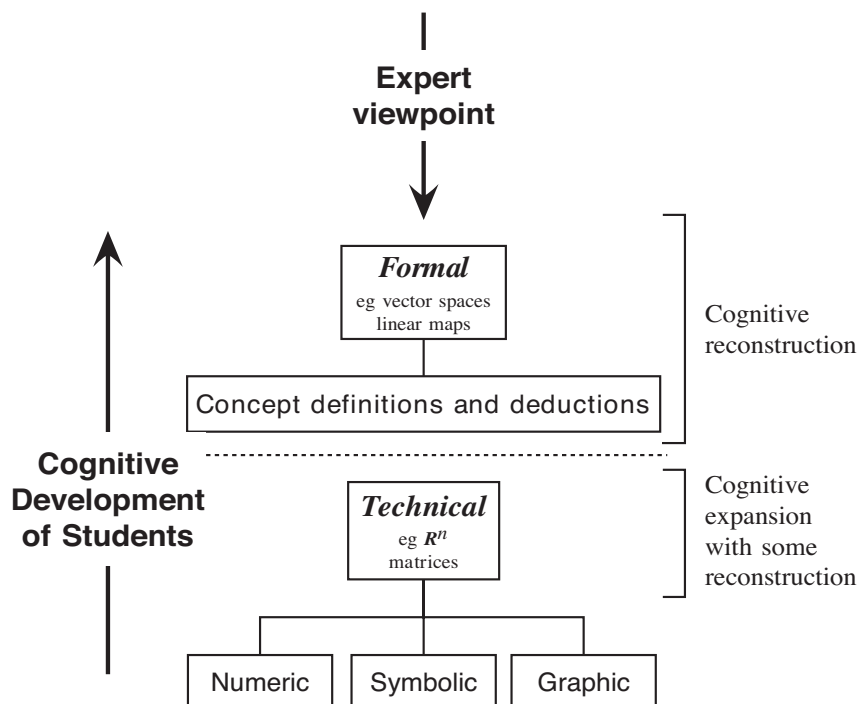
Figure 1: Two different levels of advanced mathematical thinking as exemplified in Linear Algebra

now presented to the student who must use the *definition* of a vector space from which to construct its properties. Whilst a mathematician may see this as *proving* theorems about vector spaces, the cognitive reality is different.

The student already has various intuitions about vector spaces from previous experience which may be used to *suggest* techniques of proof, but she or he must be certain that the theorem follows using only logical deductions from the definition and not from any additional intuitions. Cognitively the student must *construct* the meaning of a vector space from the concept definition. The *reconstructive generalisation* necessary to build the abstract vector space concept is cognitively far more difficult than the *expansive generalisation* needed to cope with vectors and matrices in R^n (for more detail, see Harel & Tall, 1991). My own son, who gave up mathematics after a term which included lectures on vectors, balked when faced with the definition of a vector space and took a degree in psychology instead.

Students' difficulties with the more formal aspects of mathematics are legendary both in the day-to-day experience of mathematics lectures and in the growing mathematics education literature (see Tall 1991b, 1992 for overviews).

It is interesting to see that it is far easier to program software to work at the *technical* level (for example in *MatLab*) than the *formal* level. Some software is designed to focus on formal ideas. For instance, the language *ISETL* has a syntax which is very close to set-theoretic language, as is evidenced by the following code for a function **isgroup** where isgroup(G,op)

returns TRUE if (G,op) is a (finite) group, and FALSE otherwise (Dubinsky & Leron, 1994):

```
isgroup := func(G,op);
    return (forall x,y in G | x .op y in G)
        and (forall x,y,z in G | (x .op y) .op z = (x .op (y .op z))
        and (exists e in G | (forall x in G | x .op e = x))
        and (forall x in G | (exists y in G | x .op y = e));
    end;
```

This allows experience to be obtained by working with specific groups at the technical level, whilst providing intuitions from which to build more general formal ideas.

In geometry, software such as *Cabri Géomètre* and *Geometer's Sketchpad* enable the student to construct geometric figures that can be moved around to gain intuitions to suggest theorems. But the theorems themselves still need to be proved and (unlike the example given above in *ISETL*) the sequence of steps in the formal proof may not follow obviously from the computer experience.

In the *calculus*, there is a huge array of software such as *Mathematica*, *Maple*, *Reduce*, *MACSYMA*, *Theorist*, *Derive*. Various implementations are becoming available on graphic calculators, for instance, the new TI-92 includes *Cabri Géomètre* and *Derive* environments in addition to the usual graphic, numeric and programming facilities found on other graphic calculators. Once more, however, these concentrate more on the *technical* facilities which can be programmed into the software, than on the *formal* meanings which require reflection by the individual and personal constructions from concept definitions.

When David Stoutemeyer produced *MuMath* (the predecessor to *Derive*), his main intent was to focus on the technical aspects to provide engineers with the technical support to cope with information which previously had to be looked up in extensive books of formulae. The idea that the software could be used for conceptual purposes in education was a secondary consideration which emerged later.

Physical and mental representations of mathematics

Computer software proves to be good at handling *numeric*, *symbolic* and *graphic* representations of mathematics. Numeric aspects are particularly well catered for because there are well-defined algorithms for calculation which are relatively easily programmed (as witnessed by the ubiquity of numerical calculators). Symbolic aspects have grown in prominence as the nature of the underlying symbol manipulation has been more carefully analysed and programmed with increasing success. Since the early eighties, graphic aspects have also been programmed with increasing subtlety, from the drawing of

graphs of one variable, solutions of differential equations and now on to two and three dimensional imagery of all kinds with an increasingly intuitive user-interface. These give previously unimaginable environments for *doing* mathematics and *thinking* about it.

However, the internal mental structure relating to the mathematical ideas is considerably more complex than the physical pictures or symbols we may draw on paper or on a computer screen. An extensive literature has grown out of the use of numeric, symbolic and graphic representations, using the computer to link graphs dynamically with tables of values and symbolic formulae. Graphic calculators are extensively used to draw graphs to interpret symbols and tables graphically. Software has been designed with *dynamic* links between various representations. Indeed, the spreadsheet *Excel* links tables of values to a graph which changes as the table entries change. The latter can be used for a variety of tasks such as drawing the successive values of an iteration to see the iteration converge (figure 2), or drawing the solutions of a differential equation using an iterative numerical method.
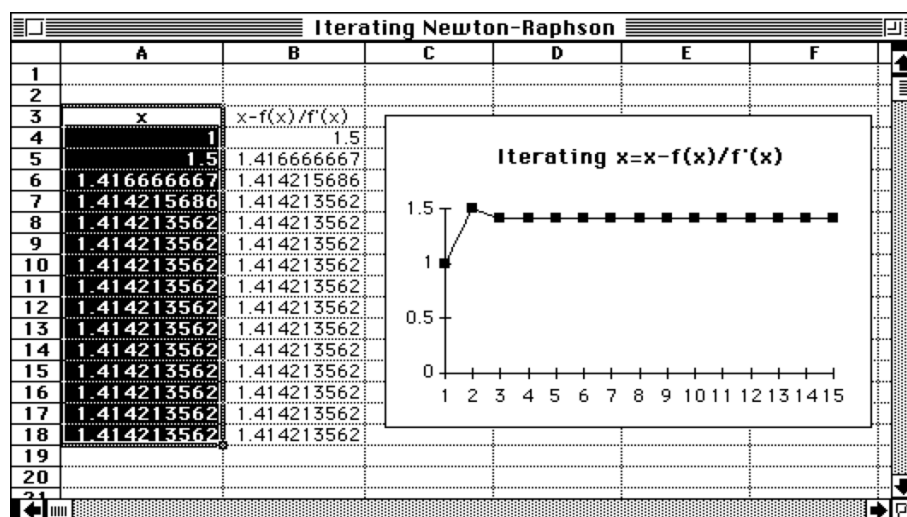

Figure 2: Using a spreadsheet to calculate Newton-Raphson approximations to a root of f(x)=x²–2

Linking symbolism to graphical representations can, however, produce unforeseen results. For instance, Caldwell (1995) expected students to find the roots and asymptotes of the rational function

$$f(x) = \frac{x(x-4)}{(x+2)(x-2)}$$

by algebraic means, only to be given a substantial number of approximate solutions such as 0.01 and 3.98 using a graphing calculator. Boers & Jones (1993) report students use of a graphic calculator to draw a graph of

$$f(x) = \frac{x^2 + 2x - 3}{2x^2 + 3x - 5}$$

4

with a removable discontinuity at *x*=1. They found that more than 80% of the students had difficulty reconciling the graph with the algebraic information, for example, drawing an asymptote suggested by the zero in the denominator, despite the graphic evidence of the calculator (figure 3).
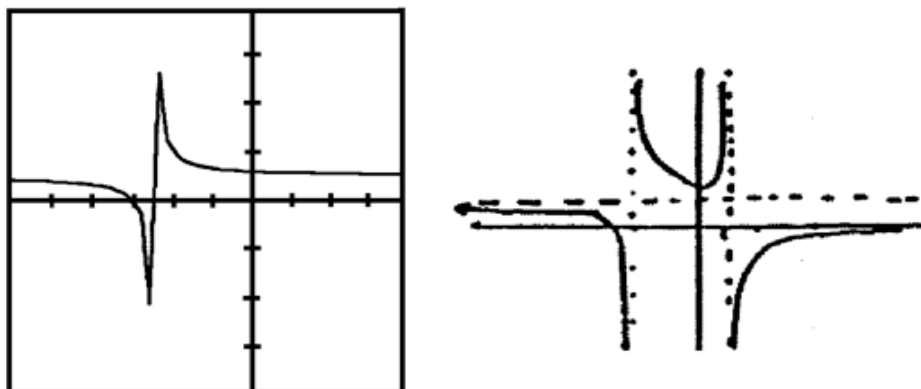

Figure 3: graphic calculator display and student graph

In both of these cases we find students moving from one representation (symbolic) to another (graphic). In the first case the students fail to relate back to the original algebra, in the second the original algebra conditions what they expect to see, and when they don't see it, they imagine it.

When multiple representations first appeared, I was unsure about the claims made about their value and too timid to go against the surge of euphoria that came with them. For me they were *intellectually* tremendously satisfying, but I had concerns that mathematicians tend not to use several representations *simultaneously*, they *shift* from one representation to another, using whichever proves the most appropriate at the time. The above examples show that students don't necessarily think in the same flexible way. More recently the euphoric tide has turned:

> Tables, graphs, and expressions might be multiple representations of functions to us, but I have seen no evidence that they are multiple representations of anything to students. In fact, I am now unconvinced that they are multiple representations even to us, but instead may be areas of representational activity among which we have built rich and varied connections. It could well be a fiction that there is any interior to our network of connections … just an expression of our sense, developed over many experiences that we can move from one type of representational activity to another, keeping the current situation somehow intact.
> (Thompson, 1993)

Seeing mathematics from *our* expert perspective and attempting to pull students "up" to it is quite different from the realities of student development. To understand the latter requires more than armchair philosophising, it requires practical observation and careful cognitive theory-building.

A spectrum of student insights into mathematics

Mathematicians have observed, anecdotally over the centuries, that some students can "do" mathematics, but others can only cope with routine problems after practice. Until recently mathematics teaching at university has concentrated on those that have a "natural talent" for mathematics, with the purpose of the degree (at least in pure mathematics) to be to produce the pure mathematicians of the future. There was a mistaken attempt in the sixties "new mathematics" to introduce set theoretic concepts into school mathematics, seeing the syllabus "from above" in the manner of experts. It didn't work. Subsequently there was a broader move to teach mathematics in a way more meaningful to children at their own level, taking into account difficulties observed in empirical research (eg Hart, 1981). But, just as the "expert view" failed the majority in the sixties, the "meaningful view for the majority" in the eighties and nineties is being seen in universities as failing the more able by omitting "difficult" topics from the syllabus, leaving students ill-prepared for university mathematics.

I suggest that the problem lies in an over-simplistic democratic view of student learning that all children go through the same stages but at different speeds. There is much research which suggests a different picture giving clearer substance to mathematician's intuitions – that different individuals think in different ways which makes some more likely to succeed than others.

Kruteskii (1976) and his co-workers (eg Dubrovna 1992, Shapiro 1992) used a wide range of mathematical problems to investigate the styles of thinking that distinguishes gifted and capable students from average or below-average students throughout primary and secondary school. For our purposes what is important is to note the following:

(a) the *capable*, and especially the *gifted*, have highly connected conceptual structures that enable them to grasp the essential elements in a problem without losing track of the details, to focus on generalities, to curtail reasoning to an essential minimum, and to think flexibly.

(b) *average* students have less well-connected conceptual structures, are more likely to focus on specifics and learn to curtail reasoning only after practice at specific procedures.

(c) the *below average* tend not to distinguish essential detail from inessential detail, they remember random specifics rather than general strategies and often break down when carrying out mathematical procedures.

(d) the *gifted*, in addition, show a spectrum of preferences for different ways of thinking; of Krutetskii's original 34 "gifted" individuals, 6 were classified as "analytic" (preferring

symbolic and logical reasoning), 5 as "geometric" (preferring visually based thinking) and 23 as "harmonic", exhibiting a spectrum of relative preferences for verbal-logical and visual thinking.

It should be emphasised that Kruteskii hypothesised that, whilst individual differences clearly exist, the qualities that make mathematical thinking powerful are teachable. My own view is more sceptical. I am convinced that all students' thinking can be improved, but the extent of the improvement is constrained by the way in which individuals process information. In particular, the different ways of thinking have long-term implications which tend to cause a divergence between success and failure as the students mature, continuing into higher education.

This is important in our study of advanced mathematical thinking, for not long ago only 10% of British students went on to study at university. In Krutetskian terms this would suggest that most of these were gifted in some way, at least well above average. Even so, experience showed that many failed to operate at a suitable powerful level in university mathematics. With the recent increase to 30% of the population now going on to higher education in Britain, many more will be classified as capable, or average. This suggests that the clientele taking university mathematics may change radically in some (but not necessarily all) universities. Different types of student almost certainly require different types of learning experience.

Current research into undergraduate mathematical learning in analysis (Pinto, in progress) suggests that the successful do not find the subject easy. On the contrary even gifted students struggle to give meaning when reconstructing their knowledge to cope with the formalism. Less successful students have a shallower view of the concepts, with intuitive, visual interpretations of limit, continuity, maximum, supremum etc.; these intuitions support the theorems stated but do not connect to the formal meaning of the definitions, so the students fail to understand the proofs.

The reasons why students end up in this position can be traced back into their earlier experiences in learning which have often degenerated into routine learning of procedures. Ausubel et al (1968) refer to the difference between *meaningful* and *rote* learning, a distinction which has been embedded in British education for several centuries:

> *Master*. … I wil propounde here ii examples to you whiche if you often doo practice, you shall be rype and perfect to subtract any other summe lightly …
> *Scholar*. Sir, I thanke you, but I thynke I might the better doo it, if you did showe me the woorkinge of it.
> *Master*. Yea but you muste prove yourselfe to do som thynges that you were never taught, or els you shall not be able to doo

> any more then you were taught, and were rather to learne by rote
> (as they cal it) than by reason.
>
> (Robert Recorde, *The Ground of Artes*, 1543)
> (quoted in Howson, 1982)

Skemp (1976) formulated a similar notion in the difference between *instrumental* and *relational* understanding, whilst Hiebert (1986) studied the difference between *procedural* and *conceptual* knowledge.

These extremes of procedural and conceptual learning are not necessarily exclusive. The procedures allow one to *do* mathematics and the conceptual relationships allow one to *think* about it. Throughout the learning of arithmetic, algebra and the calculus, the symbolism acts as a pivot to act either as a cue to carry out a procedure (such as addition) or as the output of that procedure (the sum), as in these examples:

symbol	*process*	*concept*
$3+2$	addition	sum
-3	subtract 3, 3 steps left	negative 3
$3/4$	division	fraction
$3+2x$	evaluation	expression
$v=s/t$	ratio	rate
$y=f(x)$	assignment	function
dy/dx	differentiation	derivative
$\int f(x)\,dx$	integration	integral
$\left.\begin{array}{l} \lim_{x\to 2}\dfrac{x^2-4}{x-2} \\[2ex] \sum_{n=1}^{\infty} 1/n^2 \end{array}\right\}$	tending to limit	value of limit
$\sigma \in S_n$	permuting $\{1, 2, \dots, n\}$	element of S_n

A common development in mathematics starts with gaining experience of a process, first as a specific procedure, perhaps then with more flexibility in alternative ways which are more effective or curtailed, and finally conceived as a single entity. A symbol which first evokes a process becomes seen also as the resulting concept. The use of a symbol as a pivot between **pro**cess and con**cept** is called a **procept** (Gray & Tall, 1993, 1994). It gives enormous power allowing the individual to *do* mathematics (as a process) and to *think* about it (as a concept).

The failure to think of such symbols flexibly as process and concept renders the individual unable to think in a powerful mathematical way. Instead the fall-back position is rote-learning. It may work well in routine problems, but it produces an escalating degree of difficulty in successive stages because

it is more difficult to coordinate processes than to manipulate concepts. The failing student is therefore doubly handicapped. Not only is he or she finding the mathematics difficult, the kind of mathematics he or she is forced to use is itself more difficult than the flexible thinking available to the successful mathematician.

Different kinds of procepts become successively more complex in arithmetic, algebra and calculus:

- *operational procepts*, such as those in arithmetic or symbolic differentiation with built-in algorithms of computation,

- *template procepts*, such as expressions in algebra where $2+3x$ can only be evaluated when x is given a numerical value, yet the expression itself can be manipulated as a mathematical object,

- *limit procepts*, which involve a process of getting close to a limiting value and the concept of limit.

The first of these gives the impression that mathematics is about *getting answers*, the second that it is about *manipulation of symbols*, the third causes difficulties because the limiting process no longer gives a straightforward computation to obtain the limit value. Most students sense the limit as an incomplete process rather than a limit concept. For instance, an infinite decimal is seen as a never-ending process of computation rather than a limiting value.

At various stages in elementary mathematics symbols representing processes must be rethought as mathematical concepts and the way in which it happens gets progressively more complex, so more and more students get confused and resort to rote-learning of procedures to "get answers" rather than linking ideas in a conceptual manner.

Given the input to universities of students with a great diversity of cognitive development, we therefore come to the question of how the computer might be helpful in the cognitive development of advanced mathematical thinking.

The computer in developing advanced mathematical thinking

In earlier sections we have already seen that current computer software deals with those things that can be readily programmed, including operational procepts (involving numerical calculations), template procepts (symbolic manipulation) and can do all the underlying numerical computations to draw pictures and provide an enactive interface to manipulate graphical imagery. Limit processes are dealt with more circumspectly; some have symbolically computable limits but others may only be computed to appropriate numerical accuracy. Overall, it is possible to provide software linking numeric, symbolic

and graphic representations. These will now be considered in turn and in various combinations.

Various early forms of computer software, often involving graphical representations, had a theme of "giving a sense of the concepts", rather than dealing with the underlying theoretical ideas. For instance:

> The real purpose of the program is to help the students expand their intuitions and their capacity to deal with abstractions.
> (Schwartz & Yerushalmy, *The Geometric Supposer*, 1985)

> There are many investigations, allowing one or more learners to work together, or with a teacher, to develop a mental picture of the ideas to supplement the formal manipulations. (Tall, *Graphic Calculus*, 1985a)

> Our object is to give a 'feel' for probability distributions rather than go into details of the mathematics or special applications.
> (Robinson & Bowman, *Introduction to Probability*, 1986)

> Using computer graphics *changes* a differential equations course, even at an introductory level, from rote memorization of techniques to a conceptual, visually oriented approach. … Why graphics? Because the human brain is made to process *visual* images. Because far more information can be transmitted in a picture than in a sheaf of tables. We can see many solutions *all at once*; we can see the *patterns* they form.
> (West & Noonberg, *Using MacMath in Differential Equations*, 1994)

Such pictures can inspire, but can also mislead, as in the case of drawing differential equation solutions where numerical methods may develop serious errors and the pictures may show the true solutions:

> So, you need *proof* to verify what you see. *Thinking* is essential when using graphics, there is a lot of intellectual *work*, not just pretty pictures. (West & Noonberg, 1994, p.492)

In the development of the 16-19 Curriculum for SMP I designed software for the calculus (Tall 1991a) to be used in a conceptual way linking to the corresponding numeric and symbolic ideas. My own research (Tall 1985b) showed that students were well able to link visual ideas with symbolic ideas. I proposed that students should be shown *non*-differentiable functions afrom the outset. Because a differentiable function essentially "looks straight" when highly magnified, a non-differentiable one does not: it might have a left or right gradient, it might be highly wrinkled at all magnifications. Thus visual conceptual ideas are possible at an early stage to show the breadth of possibilities.

This approach pays off in later stages where, for instance, the Newton-Raphson method may be seen in a more conceptual light. First, the students are more likely to know that the graph of a differentiable function is locally straight, so they can *see* that a tangent is a good way of approximating the graph. Better still, they may also know that the tangent can be approximated

numerically, so a numerical approximation through two close points can be used, instead of differentiating to find the theoretical tangent.

It is therefore possible to let the program step through a sequence of approximations, taking the opportunity to zoom in as required, which will reveal the locally straight graph as being virtually indistinguishable from the numerical approximation to the tangent. Furthermore, once the principle of the Newton-Raphson iteration is established it is possible to try various starting points to see what happens if one starts on a curved part of the graph away from a root. Thus questions of stability of solution depending on the starting value can be raised at the very beginning. The graph of $\sin x$ is an interesting (chaotic) case! (Figure 4.)
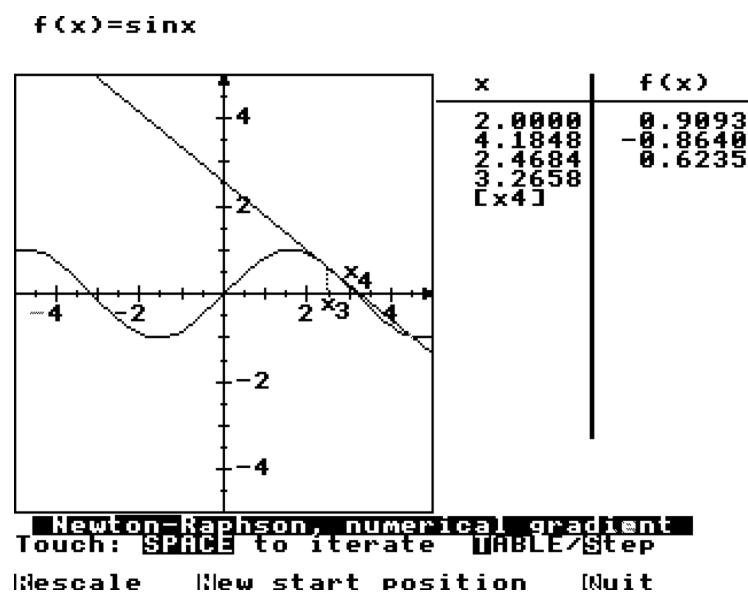

Figure 4 : Tabulating the Newton-Raphson iteration

Very few other curricula have followed such a conceptual lead. For instance, the Harvard Consortium decided that

> … we should be showing students the power of the calculus, not the special cases where it fails. …if we teach Newton's method, we should not emphasize the cases where it doesn't work.   (Sudholz, 1995, p. 57)

Instead, those that use computers in the calculus broaden concepts a little (for instance by using piecewise defined functions) but they generally fail to grasp the conceptual nettle to show not only where the calculus works, but also where it may fail. In terms of sales, the most conceptually radical course (Dubinsky et al, 1995) based on a theory of cognitive development, sold 1,200 copies in the fall of 1994, *Calculus with Mathematica* (Davis et al, 1994) using *Mathematica* notebooks, sold 4,000 copies, whilst the text of the Harvard Consortium mentioned above sold 60,000 copies. (Further details of these and other calculus reform projects can be found in Tucker & Leitzel, 1995).

Powerful combinations of visual and symbolic

Early research showed that it was possible to use symbol manipulators to carry out routine manipulation successfully whilst using graphical software for visual conceptualisation.

Palmiter (1991) taught integration to students using the symbolic software *MACSYMA* to perform all the computations for five weeks whilst a comparison group were taught the same course without a computer for ten weeks. At the end of the course both sets of students took one exam on conceptual ideas and another on techniques under identical conditions, except that in the techniques exam the *MACSYMA* students had half the time but could use the symbol manipulation software. The results showed a significant improvement in the students using the computer over those without:

| Examination | Class | | T^2 | $p<?$ |
	MACSYMA	Traditional		
Conceptual	89.8% (±15.9)	72.0% (±20.4)	1.20	$p<0.001$
Computational	90.0% (±13.3)	69.6% (±24.2)	0.92	p<0.001

Although students using symbol manipulation software are likely to get far less practice at routine manipulation, some authors suggest that this need not prevent them from developing the facility to carry out the manipulations by hand. Using the early symbol manipulation software *MuMath* for twelve weeks with just three weeks to practice techniques:

> Students showed deep and broad understanding of course concepts and performed almost as well on a final exam of routine skills as a group who had studied the skills for the entire fifteen weeks.
>
> (Heid, 1984, p. 2; 1988)

When students studied calculus using *Mathematica*, focusing on conceptual ideas and using the software to carry out the computations, it has been claimed that:

> Early evaluations indicate that students have no disadvantage in advanced courses based on hand calculations.
>
> (Davis, Porta & Uhl, 1992, p. 311)

Such students may grow to see mathematics in a new light:

> I was helping a friend in the normal (read: old-fashioned, obsolete) section, and I worked a problem down to the integral and stopped there, satisfying myself that a computer could take it from there. My friend looked at me, stunned that I had not done the hardest part of the problem and considered myself finished. I was finished, for I had done the thinking behind the problem and didn't want to bother myself with the petty details of working through memorized procedures. That's what a computer's for.(ibid, p. 310)

On the other hand, there is evidence that focusing on certain aspects and neglecting others may cause the neglected items to atrophy. For instance, sixth-form students using *Derive* on hand-held computers to draw graphs of functions did not need to substitute numerical values for the independent variable to get a table of values to draw the graph. As a result, they had little practice of numerical substitution. This had unforeseen consequences. Some students who could calculate by substitution before the course were unable to do so afterwards. The students were asked:

"What can you say about *u* if *u=v+3*, and *v=1*?"

None of the seventeen students improved from pre-test to post-test and six successful on the pre-test failed on the post-test (Hunter, Monaghan & Roper, 1993).

Symbol manipulators can also give new kinds of meaningless procedures for (weaker?) students who may rote-learn sequences of key-strokes in place of rote-learnt symbolic procedures. For instance, *Derive* replaces the procedure of symbolic differentiation by a sequence of keystrokes:

- select **Author** and type in the expression,
- select **Calculus**, then **Derivative**,
- specify the variable (e.g. **x**),
- **Simplify** the result.

In a comparison of two schools in the UK, one following a standard course, one using *Derive*, the students were asked:

Please explain the meaning of $\lim_{h \to 0} \dfrac{f(x+h) - f(x)}{h}$.

… All the students in the school [following the standard course] gave satisfactory theoretical explanations of the expression but none gave any examples. However, none of the *Derive* group gave theoretical explanations and only two students [out of seven] mentioned the words 'gradient' or 'differentiate'. Four of the *Derive* group gave examples. They replaced *f(x)* with a polynomial and performed or described the sequence of key strokes to calculate the limit.

(Monaghan, Sun & Tall, 1994.)

Failure of some computer approaches to the limit concept

It is well-known (Monaghan, 1986) that students (and professors) have idiosyncratic views of the nature of the real line. In fact there is a Catch-22 situation where real numbers cannot be understood without limit concept, nor limits fully appreciated without real numbers. Programming in a computer language or numeric system with limited accuracy seems to make this worse. For over fifteen years my department has used a course in BASIC programming designed to give students intuitions to form a basis for the later formalities of mathematical analysis. Li & Tall (1993) investigated this idea

and found it flawed. The calculation of the sum of one series took time so that, although 10 or 100 terms would be computed almost instantaneously, 1000 terms might take longer and 10000 terms ten times as long again. The result is that the students sensed that the process would never end, because the bigger the number, the longer it took. The net result was that half the students believed that an increasing sequence bounded above may simply continue creeping up without ever converging to a limit. These *refused* to accept the completeness axiom because they did not believe it! (which is not really so strange because they had little real understanding of numbers beyond the rationals which fail to be complete anyway...)

These students were intending teachers following a mathematics education course who were capable rather than gifted. They illustrate the kind of difficulties that such students may have. There were some changes in their understandings during the course, including subtle modifications of the meaning of terms such as "tends to" and "limit":

Complete the following sentences:
1, 1/2, 1/4, 1/8, …   tends to _____
The limit of 1, 1/2, 1/4, 1/8, is _____

"tends to" / "limit"	$0/0$	$0/\dfrac{1}{\infty}$	$\dfrac{1}{\infty}/\dfrac{1}{\infty}$	$0/?$	$2/2$	$0/2$	$0/1$
pre-test (*N*=25)	0	**11**	1	5	0	2	2
post-test (*N*=23)	**8**	3	3	0	4	0	2

But despite a lecture attempting to give meaning to an infinite decimal, the decimal "0·9 repeating" did not change its image:

Is $0\cdot\dot9 = 1$	Y	N	?	no response
pre-test (*N*=25)	2	**21**	1	1
post-test (*N*=23)	2	**21**	0	0

Interviews revealed that students continued to conceive $0\cdot\dot9$ as "a sequence of numbers getting closer and closer to 1", or not a fixed value "because you haven't specified how many places there are" or "it is the nearest possible decimal below 1".

Two other questions gave further insight into decimals as processes:

(A) Can you add $0\cdot1+0\cdot01+0\cdot001+\cdots$ and go on forever to get an exact answer? *Y/?/N*
(B) $1/9 = 0\cdot\dot1$ Is 1/9 equal to $0\cdot1+0\cdot01+0\cdot001+\cdots$? (*Y/?/N*)

The favoured response on both pre-test and post-test is *No* to (A) and *Yes* to (B):

Responses to (A)/(B)	Y / Y	Y / N	N / N	**N / Y**	N / ?	? / N	nr / Y
pre-test (*N*=25)	4	0	1	**18**	0	1	1
post-test (*N*=23)	**2**	2	2	**14**	1	0	0

Why do the majority regard $0 \cdot 1 + 0 \cdot 01 + 0 \cdot 001 + \cdots = 1/9$ as *false* but $1/9 = 0 \cdot 1 + 0 \cdot 01 + 0 \cdot 001 + \cdots$ to be *true*. If these sentences are read as *processes* rather than equality of objects, the first is a potentially infinite process which can never be completed but the second shows how 1/9 can be divided out to get as many terms as are required. In this way computer experiences may fail to dislodge a firmly held intuitive belief.

Effects of Computer Algebra Systems on Mathematical Proof

Symbol manipulators focus on getting results rather than taking the user through the process of manipulation to obtain the result. As such they focus on processes that *work* rather than processes that *fail*. Thus counter-examples are few and the *need* for general proof is not evident. Virtually every graph ever drawn is of a differentiable function. Efforts to use absolute values and integer parts to motivate non-differentiability and discontinuity, are regrettably not typical non-examples. Thus it is that computer algebra software continues to focus on the technical aspects of mathematics without little need for formal proof.

Software can be designed for motivating more general concepts. for instance, the software written for the SMP 16-19 syllabus for the Archimedes computer includes a facility to model functions that take different values on rationals and irrationals and, with a little insight, software like *Mathematica* can give visual support to more interesting non-examples (e.g. Rosenthal, 1992 who draws graphs of functions where f(p/q)=1/q for a rational p/q in lowest terms).

The software written for the Archimedes computer for SMP 16-19 is, of course, a cheat, because *all* computer numbers, (especially 8-digit representations) are rational. However, by computing the continued fraction approximation to a number *x* and declaring it to be "pseudo-rational" if the denominator grows suitably large can separate computer numbers into two disjoint subsets which are intimately entangled. In the case of the SMP software, a randomly generated number has a 94% chance of being "pseudo-irrational", and this offers the possibility of discussing what happens with randomly Riemann sums which tend to alight far more often on pseudo irrationals. Figure 5 shows the area under the graph whose values are *x* (*x* rational) and 1–*x* (*x* irrational). Using a rational step length and the mid ordinate rule picks up only rational values. Using a random length picks up mainly (pseudo)-irrationals. By using double precision, the model can be fixed to give 99.99% pseudo-rationals, but I enjoy the inadequacy of the model in discussing ideas of randomness and irrationals with students. This nicely leads into an intuitive discussion of the Lebesgue integral!
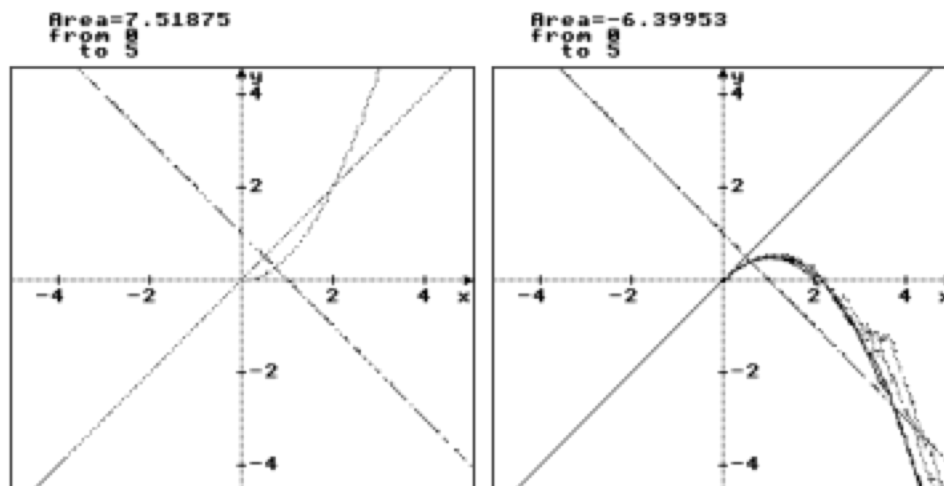
Figure 5 : calculating the area under a graph with rational and random steps

Reflections

Thinking about the empirical evidence gleaned from various educational studies, it becomes clear that there is much to do to understand the effects of the computer on undergraduate learning. Computer software is usually programmed to operate at a technical level rather than a formal level. It usually carries out the processes internally giving the student a new and not necessarily better view of the balance between process and concept, in which symbolic procedures may be replaced by sequences of key presses. But it has the potential to be used imaginatively to give all kinds of conceptual insights.

In practice, when the need to complete the syllabus presses upon those giving courses, conceptual insights are often suppressed in favour of traditional concepts given a token computer support. However, as the computer is used more by mathematicians, conceptions of traditional mathematics will begin to change to fit the opportunities of the new medium. Proof will remain the ultimate test of mathematical viability, but hypothesis and computer experiment will grow.

It remains important that even as new opportunities are taken by mathematicians in using the computer that attempts are made to see mathematics from a growing cognitive viewpoint of student learning. This will require careful experiment, observation and deep reflection about what is happening in undergraduate learning, cognisant of the broad spectrum of needs and learning styles to be found in our students.

References

Ausubel, D. P., Novak, J. D. & Hanesian, H. (1968): *Educational Psychology: A Cognitive View*, Holt, Rinehart & Winston.

Boers, M, & Jones, P. (1993). Graphics calculators and calculus: Impact on student attitude and practice. In M. Artigue & G. Ervynck (eds.), *Proceedings of the Working Group 3 on Students' Difficulties in Calculus*, ICME-7, Québec, Canada, 84–86.

Caldwell, J. H. (1995). Assessment and graphing calculators. In L. Lum (ed.) *Proceedings of the Sixth Annual International Conference on Technology in Collegiate Mathematics*, (pp. 99–105). Reading MA: Addison-Wesley.

Davis, B., Porta, H. & Uhl, J. (1991). *Calculus & Mathematica*. Reading MA: Addison-Wesley.

Davis, B., Porta, H. & Uhl, J. (1992). Calculus & Mathematica: addressing fundamental questions about technology, *Proceedings of the Fifth International Conference on Technology in Collegiate Mathematics*, (Addison-Wesley), 305–314.

Dubinsky, E. & Leron, U.: 1994. *Learning Abstract Algebra with ISETL*, Berlin: Springer-Verlag.

Dubinsky, E., Schwingendorf, K. E. & Mathews, D. M. (1995). *Calculus, Concepts & Computers* (2nd edition) New York: McGraw-Hill.

Dubrovna, I. V. (1992). A study of mathematical abilities in children in the primary grades. In Kilpatrick, J. (Ed.), *Problems in the Psychology of Abilities: A Collection of Articles*, (pp. 3–64), Chicago: UCSMP.

Gray, E. M. & Tall, D. O. (1993). Success and Failure in Mathematics: The Flexible Meaning of Symbols as Process and Concept, *Mathematics Teaching*, **142**, 6–10.

Gray, E. M. & Tall, D. O. (1994). Duality, Ambiguity and Flexibility: A Proceptual View of Simple Arithmetic, *The Journal for Research in Mathematics Education*, 26 (2), 115–141.

Harel, G. & Tall, D. O. (1991). The General, the Abstract, and the Generic in Advanced Mathematics, *For the Learning of Mathematics*, **11** 1, 38–42.

Hart, K. M. (Ed.), (1981): *Children's Understanding of Mathematics: 11–16*, John Murray, London.

Heid, K. (1984): *Resequencing skills and concepts in applied calculus through the use of the computer as a tool*, Ph.D. thesis, Pennsylvania State University.

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

Hiebert, J. (Ed.), (1986) *Conceptual and procedural Knowledge: The Case for Mathematics*, Hillsdale. NJ.: Erlbaum.

Howson, A. G. (1982). *A History of Mathematics Education in England*. Cambridge: CUP.

Hunter, M., Monaghan, J. D. & Roper, T. (1993). The effect of computer algebra use on students' algebraic thinking. In R. Sutherland (Ed.), *Working Papers for ESRC Algebra Seminar*, Institute of Education, London.

Krutetskii, V. A. (1976). *The Psychology of Mathematical Abilities in Schoolchildren* (Trans, J. Teller, ed. J. Kilpatrick & I. Wirzup). Chicago IL: University of Chicago.

Li, L. & Tall, D. O. (1993). Constructing different concept images of sequences and limits by programming. *Proceedings of PME XVII*, Tsukuba, Japan, II, 41–48.

Monaghan, J. D. (1986). *Adolescent's Understanding of Limits and Infinity*. Unpublished Ph.D. thesis, Warwick University, UK.

Monaghan, J. D., Sun S. & Tall, D. O. (1994). Construction of the limit concept with a computer algebra system, *Proceedings of PME XVIII*, Lisbon, III, 279–286.

Osterbee, A. & Zorn, P. (1994). *Calculus from Graphical, Numerical, and Symbolic Points of View* (preliminary edition), Fort Worth: Saunders.

Palmiter, J. R. (1991). Effects of Computer Algebra Systems on Concept and Skill Acquisition in Calculus, *Journal for Research in Mathematics Education*, 22 2,151–156.

Robinson D. A. & Bowman A. W. (1986). *Introduction to Probability*, (for the BBC and IBM computers), Bristol & Boston: Adam Hilger.

Rosenthal, J. W. (1992). Seeing is Believing in Analysis? *Proceedings of the Fifth Annual International Conference on Technology in Collegiate Mathematics*. Addison-Wesley, 437–452.

Schwartz J. and Yerushalmy M. (1985). *The Geometric Supposer: Triangles* (for the Apple Computer). Pleasantville, NY: Sunburst.

Shapiro, S. I. (1992). A psychological analysis of the structure of mathematical abilities in grades 9 and 10. In Kilpatrick, J. (Ed.), *Problems in the Psychology of Abilities: A Collection of Articles*, (pp. 97–142), Chicago: UCSMP.

Skemp, R. R. (1976). Relational understanding and instrumental understanding, *Mathematics Teaching* 77, 20–26.

Sudholz, H. O. (1995). *The Calculus Consortium based at Harvard University*. In A. C. Tucker & J. R. C. Leitzel, *Assessing Calculus Reforms* (p. 57.) MAA Report.

Tall, D. O. (1985a). *Graphic Calculus*, ABCo Ltd: London.

Tall, D. O. (1985b). *Building and Testing a Cognitive Approach to the Calculus using Interactive Computer Graphics*, Ph.D. Thesis in Education, Warwick.

Tall, D. O. (1991a). *Real Functions & Graphs: SMP 16-19* (computer programs for the Archimedes computer), Cambridge University Press,

Tall, D. O. (1991b). *Advanced Mathematical Thinking* (editor), Kluwer: Dordrecht.

Tall, D. O. (1992). The Transition to Advanced Mathematical Thinking: Functions, Limits, Infinity, and Proof, in Grouws D. A. (ed.) *Handbook of Research on Mathematics Teaching and Learning*, Macmillan, New York, 495–511.

Thompson, P. W. (1993). Students, Functions, and the Undergraduate Curriculum. Presentation to the Annual Joint Meeting of AMS/MAA. January, 1993. To appear in *Contemporary Mathematics*, (publ. American Mathematical Society).

Tucker, A. C. & Leitzel, J. R. C. (1995). *Assessing Calculus Reforms*. MAA Report.

West, B. & Noonberg (1994), Using *MacMath* in Differential Equations. *Proceedings of the Fifth International Conference on Technology in Collegiate Mathematics*, (Addison-Wesley), 490–495.