Biological Brain, Mathematical Mind & Computational Computers

(how the computer can support mathematical thinking and learning)

David Tall
Mathematics Education Research Centre
University of Warwick, UK
e-mail: David.Tall@warwick.ac.uk

This presentation considers how the peculiar structure of the biological brain may be supported by the computational power of the computer to enhance mathematical thinking. It considers how we think and learn mathematics with particular reference to the use of visualisation and symbol manipulation. Visualisation occupies a major portion of the brain's cortex and enables Homo Sapiens to 'see' how ideas can be formed and related.

Mathematical symbols in arithmetic, algebra, calculus particularly suit the biological brain, acting as pivots between concepts for thinking about mathematics and processes to calculate and predict. We use the term 'procept' to describe this particular combination of symbol as process and concept. Analysis of procepts reveals that the development of symbols does not follow an easy cognitive path for the growing individual because they operate in significantly different ways in arithmetic, algebra and the calculus. We therefore advocate a versatile approach that complements the visualisation of concepts with the power of symbolic calculation to model, calculate and predict. Empirical evidence is provided to show how theory relates to practice when the computational computer is used to enhance and develop the power of the mathematical mind.

Figure 1: brain, mind, computer

INTRODUCTION

After several million years of humanoid evolution and a hundred thousand years of *Homo Sapiens*, the development of the computer has taken but a few decades. The carbon-based brain and silicon-based computer work in very different ways. It is the purpose of this presentation to relate the ways in which the two systems work to enhance the mathematical thinking processes of the individual. I will use the term *mathematical mind* to refer to the manner in which the processes and concepts of mathematics are conceived and shared between individuals. The discussion of the mathematical mind will focus both on the mathematical structure that is overtly shared by the mathematical community and also on the underlying manner in which our biological mental structure handles these ideas. First therefore we consider what is known of the biological operation of the brain, build on this to develop theories of the mathematical mind, then consider how this interacts with the computer in developing and enhancing mathematical thinking. This discussion will be supported by evidence from empirical research in using the computer in mathematics education in a variety of different ways. In particular, we will consider how the computer may be used to enable the brain to operate in a versatile manner using visual imagery to complement the manipulation of symbols.

BRAIN, MIND AND COMPUTER

The Biological Brain

The last decade of the last millennium was dedicated as 'the decade of the brain'. During this period, those interested in research into brain activity made a concerted effort to understand how the brain works and to make their findings available to a wider public. Some of this research was directed towards understanding in mathematics, often focused on empirical studies of brain activity in arithmetic, such as *The Number Sense* by Stanislas Dehaene (1997), or *The Mathematical Brain* by Brian Butterworth (1999). These and a number of other popular books on the subject (Edelman (1992), Pinker (1997), Greenfield (1997), Carter (1999), Devlin, (2000)) offer tantalizing insights into the structure and operation of the biological brain. For instance, a newborn child already has the mental structure for a primitive numerosity to cope with numbers up to three based on its built-in ability to identify and track a small number of objects. But, as Dehaene has responded privately by e-mail, the possibility of getting actual physical evidence of how the brain copes with higher mathematical constructs is still in the future. In building up a theory of mathematical mind, we therefore rely on theoretical constructs that are designed to correspond to what is known about the way in which the biological brain works.

The brain is a complex multi-processing system. To simplify thinking processes, regular activities are routinised so that they require less brain activity and we do not have to attend to them consciously:

> As a task to be learned is practiced, its performance becomes more and more automatic; as this occurs, it fades from consciousness, the number of brain regions involved in the task becomes smaller. (Edelman & Tononi, 2000, p.51)

The remaining complexity is made more manageable by suppressing less important detail:

> The basic idea is that early processing is largely parallel – a lot of different activities proceed simultaneously. Then there appear to be one or more stages where there is a bottleneck in information processing. Only one (or a few) "object(s)" can be dealt with at a time. This is done by temporarily filtering out the information coming from the unattended objects. The attentional system then moves fairly rapidly to the next object, and so on, so that attention is largely serial (i.e., attending to one object after another) not highly parallel (as it would be if the system attended to many things at once). (Crick, 1994, p. 61)

Thus routine processes are performed subconsciously while the conscious 'focus of attention' or 'short-term working memory' attends to the important aspects. This can be made even more efficient by making the conscious elements as 'small as possible'. This can be done using words, or even more efficiently, symbols:

> I should also mention one other property of a symbolic system – its compactibility – a property that permits condensations of the order $F=MA$ or $S=\frac{1}{2}gt^2$, …in each case the grammar being quite ordinary, though the semantic squeeze is quite enormous. (Bruner, 1966, p. 12.)

It is by a combination of these phenomena that it becomes possible for the biological brain to become a mathematical mind.

Mathematical Mind

During the last two decades my colleagues and I have introduced a number of constructs to describe and explain the cognitive operation of the mathematical mind. These include:

- the ***concept image***, which refers to the total cognitive structure in an individual mind associated with the concept, including all mental pictures, associated properties and processes (Tall & Vinner, 1981),
- a theory of ***cognitive units*** (the mental chunks we use to think with, and their related cognitive structure). (Barnard & Tall, 1997).

A particular type of cognitive unit which is highly important in mathematics is:

- the notion of ***procept***, referring to the manner in which we cope with symbols representing both mathematical ***pro***cesses and mathematical con***cepts***. (Gray & Tall, 1994). Examples include 3+5, ax^2+bx+c, $\frac{d}{dx}\left(e^x \sin x\right)$, or $\sum_{n=0}^{\infty} \frac{x^n}{n!}$.

The notion of *concept image* has proved a most valuable construct in contrasting the activities of the mathematical mind and the computer. Whilst the computer is confined to the algorithms to carry out computations and represent solutions in numbers and pictures, the mathematical mind has all kinds of associations within the multi-processing brain. For instance the concept of 'function' not only has the properties given by the definition, its image includes links to a range of other ideas. These may include the notion that a function must be given by a single formula, its graph is smooth, it has a domain consisting of all the points where the function is defined, and its graph has a recognisable shape (such as those of polynomials, trigonometric expressions, exponentials and logarithms) (Vinner, 1983, Bakar & Tall, 1992).

The notion of *procept* and the more general notion of *cognitive unit* are powerful ways of describing how the mathematical mind handles mathematical concepts. Procepts can be seen in action when a child calculates 8+6 as 8+2+4=10+4=14, with the procept 6 seen equivalently as 2+4, then the expression 8+2+4 seen in a different way as 8+2 (=10) and 4, giving 14.

Symbols as procepts therefore operate in a manner that is ideally suited to the biological brain. They can be used as tokens that are easily accommodated by the limited focus of attention, and they can also be used to evoke processes that are performed largely subconsciously, leaving the conscious attention to focus on more important matters.

The brain is not naturally configured for rapid and efficient arithmetic (Dehaene, 1997). Instead it often uses meaningful links between cognitive units. For instance, when my colleague, Shaker Rasslan was discussing an algorithm for divisibilty by seven, I found myself responding not in terms of the division algorithm but in terms of associated number facts. He asked me if 121 is divisible by seven, I replied 'no, because it is eleven squared', evoking not only the factors of 121, but implicitly appealing to uniqueness of factorization.

I replied to the divisibility of 131 by 7 saying 'no, because it is 140 take away 9'. The number 119, caused me a problem. I associated it with the non-divisibility of the nearby number 121 but, of course this is of no help. I resorted finally to the division algorithm to find 119 divided by 7 is 17. However, my mind did not stop there, I saw new relationships—17 times 7 is 10 sevens and 7 sevens, which is 70+49=119, or it is 20 sevens take away 3 sevens, which is 140–21=119. I was happy once more. My mathematical mind had perceived patterns that 'made sense' to me, using the rich (yet limited) concept image I had developed for the number 7.

Mathematical thinking is therefore more than knowing procedures 'to do'. It involves having a knowledge structure which is compatible with the biological structure of the human brain, with its huge store of knowledge and internal links, and its way of coping with the profusion of simultaneous activities by using a manageable focus of attention.

In *The Humane Interface* (2000), Jeff Raskin (the designer of the Macintosh Computer) refers to this as the *locus* of attention, to indicate that the choice of items to attend to may not always result from a conscious decision. He suggests that the locus of attention can only focus on *one* thing at a time. Whilst a case can be made for this, it is not always evident how many different things are in focus simultaneously. For instance, the focus may be on the *relationship* between two different things. This may be calculated as *one* thing (the relationship), but it also requires close access to the two entities being related, a total of *three* things—the two entities and the relationship between them. In my own theoretical formulation, I therefore consider that the focus of attention concentrates on a small number of items (cognitive units) at a time. Powerful thinking arises out of the use of linkages both *within* the units themselves and *between* the units.

Consider, for example, the notion of 'linear relationship' between two variables. This might be expressed in a variety of ways

- an equation in the form $y=mx+c$,
- a linear relation $Ax+By+C=0$,
- a line through two given points,
- a line with given slope through a given point,
- a straight-line graph,
- a table of values,

and so on. Crowley (2000) (reported in Crowley & Tall, 1999) reveals how successful students develop the idea of 'linear relationship' as a rich cognitive unit encompassing most of these links as a single entity, whilst the less successful simply carry around a 'cognitive kit-bag' of isolated tricks to carry out specific algorithms. The cognitive kit-bag may get the student through the examination, but it is too diffuse to build on in later courses and students may soon reach a point where the ideas they are handling place too great a cognitive burden, leading inexorably to failure.

Computational computers

The computer is quite different from the biological brain and therefore can be of value by providing an environment that *complements* human activity. Whilst the brain performs many activities simultaneously and is prone to error, the computer carries out individual algorithms accurately and with great speed. Computer calculations with numbers and manipulation of symbols has some similarities with the notion of procept. Internal computer symbolism is used both to represent data and also to perform routines to manipulate that data. However, there are significant differences. The computer is simply a device which manipulates information *in a way specified by a program*. It has none of the cognitive richness (or baggage) of the concept image available to the human mind to guide (or confuse) problem-solving activities.

The simplest facilities involve the programming of the four rules of arithmetic, now readily available everywhere on hand-calculators. More sophisticated use of evaluations of algebraic expressions provide software environments such as spreadsheets to carry out desired calculations using given values in various cells.

The manipulation of symbolic expressions is more subtle. Not all that long ago, a colleague of mine, an eminent mathematician, saw the processes of solving equations and performing the rules of calculus required the intelligence of a human mind to operate them. He told me he had "a zoo of functions and techniques" in his head and he selected intelligently from them to perform the operations of symbolic calculus. He was amazed with the arrival of computers and the realisation that symbol manipulation could be reduced to the level of mechanical algorithms. Of course, they are different algorithms from those of arithmetic. The simplification of expressions involves as allowing an expression such as X*(–Y) to be replaced by the expression –(X*Y) where X and Y are themselves expressions. Symbol simplification utilises a list of interchangeable symbols recursively to obtain a simpler equivalent result. Formulating the appropriate list is non-trivial and subsequent versions of software have shown successive improvements. For instance, an early version of *Derive*, simplified

```
            n   n                         n     n
 (x+h) -x                         (x+h)     x
 ---------          to give       ----- - ---  .
     h                              h       h
```

and the limit option applied to this expression, as h tends to 0, gave not nx^{n-1}, but

```
  n LN(x)-LN(x/n)
ê                 .
```

The current version of *Derive* has added a new rule to simplify this expression further to give

```
    n–1
n x     .
```

The internal handling of symbols needs to be performed as a finite algorithm and can be quite different from those used by the mathematical mind. For instance, whereas a mathematician might compute a limit such as

$$\lim_{x\to 0}\frac{\sin x}{x}$$

from first principles, perhaps using a visual argument supported by a concept image of the graph of $\sin x$, a symbol manipulator is likely to use L’Hôpital’s rule to get

$$\lim_{x\to 0}\frac{\sin x}{x}=\lim_{x\to 0}\frac{D(\sin x)}{D(x)}=\frac{\cos 0}{1}=1.$$

The person using the software may have no idea of the internal mechanisms so a learner is unlikely to build up proceptual relationships without other experiences.

Relationships between Brain, Mind and Computer

Given the constraints and support in the biological brain, the concept imagery in the mathematical mind can be very different from the working of the computational computer. A professional mathematician who has mathematically formed cognitive units may use the computer in a very different way from the student who is meeting new ideas in a computer context. For example, mathematicians can make insightful use of symbol manipulators in all kinds of ways to support their already formed mathematical imagery. But students using such software have to make sense of what is going on using only their internal cognitive structure and the external guidance of the teacher.

Focusing on certain aspects of an activity and neglecting others may cause the neglected items to atrophy. Hunter, Monaghan & Roper (1993) observed that 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.

Furthermore, students see the mathematics through the embodied actions they perform within their context of action. A symbol manipulator replaces the mathematical procedures of differentiation by the selection of a sequence of procedures in the software. For instance *Derive* requires the user to take the following sequence of decisions carried out by touching the appropriate keys:

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

What happened in a comparison of two schools in the UK, one following a standard course, one using *Derive* is as follows:

> Please explain the meaning of $\lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$.
>
> … All the students in school *A* 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.)

The result is that the students using the symbol manipulator saw differentiation as a sequence of keystrokes applied for a specific symbolic expression, rather than a conceptual idea of ‘rate of change’.

To consider how the computer is used to develop conceptual insight into mathematical concepts, we need to first consider the manner in which the mathematical mind handles the development of symbols in the mathematics curriculum.

COGNITIVE DEVELOPMENT OF SYMBOLS

The mathematical mind uses symbols in a way that is dictated by the workings of the underlying biological brain. Although the mathematical development from arithmetic through algebra, calculus and on to axiomatic mathematics seems to be coherent and logical in a mathematical sense, the cognitive development is more complex. For example, all procepts have ingredients that allow the individual to switch between process and concept, however, they behave in very different ways:

(1) ***arithmetic procepts***, 5+4, 3×4, $\frac{1}{2}+\frac{2}{3}$, 1·54÷2·3, all have built-in algorithms to obtain an answer. They are *computational*, both as processes and even as concepts. For instance in the sum 8+6, the concept 6 can be linked to the operation 2+4, which can be combined in the sum 8+2+4 to give 10+4 which is 14.

(2) ***algebraic procepts***, e.g. 2+3x, can only be evaluated if the value of x is known. Thus an algebraic procept has only a *potential process* (of numerical substitution) and yet the algebraic expressions themselves represent manipulable concepts (manipulated using algebraic rules of equivalence).

Meaningful power operations such as

$$2^3 \times 2^2 = (2 \times 2 \times 2) \times (2 \times 2) = 2^5$$

can act as a cognitive basis for the power law

$$x^m \times x^n = x^{m+n}$$

valid for all real x and for whole numbers m, n. This then leads to a new use of symbols as procepts:

(3) ***implicit procepts***, such as the powers $x^{1/2}$, x^0 or x^{-1}, for which the original meaning of x^n no longer applies. (For instance, we can hardly speak of 'half an *x* multiplied together', or 'no *x*s multiplied together'— surely on *x*s must give zero— or even 'minus one *x*s'.This is as foolish as talking about 'minus one cows'.) Many students are confused by this use of symbolism that has—for them—no meaning. The lucky few see that the power law can be *generalised* and used as an axiomatic basis for deduction. Thus, for $m = n = 1/2$, we get

$$x^{1/2} \times x^{1/2} = x^1 = x$$

from which we may deduce that $x^{1/2} = \sqrt{x}$. Some find this an attractive and appealing generalisation. But for many it is meaningless. The meaning is being deduced from a law that they do not *know* is true from their experience of the world, so for them it is confusing to base their deductions on something they do not understand.

(4) ***limit procepts***, such as $\lim_{x \to a} \frac{x^3 - a^3}{x - a}$ or $\sum_{n=1}^{\infty} \frac{1}{n^2}$ have ***potentially infinite processes*** 'getting close' to a limit value, but this may not be computable in a finite number of computation.

(5) ***calculus procepts***, such as $\frac{d(x^2 e^x)}{dx}$ or $\int_0^{\pi} \sin mx \cos nx \, dx$ are more familiar in the sense that they (may) have finite operational algorithms of computation (using various rules for differentiation and integration).

These successive changes in meaning can cause serious discontinuities in building up knowledge. Success in arithmetic accrues to those who build up rich cognitive units for number concepts, failure is far more likely to result for students who remain entrenched in the procedures of counting (Gray & Tall, 1994). The move to algebra proves difficult for students who see the symbols only as processes of numerical evaluation and are less able to think of expressions as manipulable concepts. Davis, Jockusch and McKnight (1978) noted the difficulty seventh-graders have in understanding a symbol such as '7+*x*', as they complain, 'But how can I add 7 to *x*, when I don't know what *x* is?'

This can lead to attempts at coping through rote-learning of rules which may then be applied in the wrong context. For instance, the rule to compute $3a^2 \times 4a^3$ to give $12a^5$ by 'multiplying numbers and adding powers' may be used inappropriately in arithmetic to compute $3^2 \times 4^3$ as 12^5, or the idea that a fraction such as $\frac{12}{6}$ is correctly computed by dividing 12 by 6 is misapplied to $\frac{a^{12}}{a^6}$, by seeing the division bar as an instruction to divide the numbers to get a^2 (Anderson, 1997).

Limit procepts cause widespread problems because they are seen as an unfinished *process* rather than a completed *concept*. Thus the sequence $1 + 1/2 + 1/4 + \ldots + 1/2^{n-1} + \ldots$ is seen not as representing a fixed number but as a growing quantity which is 'just less than 2'. Likewise, the infinite decimal 'nought point nine repeating' is seen as 'just less than one' because, 'no matter how many places you take, it is never actually *equal* to one.' In this way students gain concept images of limits as being 'arbitrarily close' or 'arbitrarily small' or 'arbitrarily large', envisaging a number system which has infinitesimals and infinite quantities within it (Cornu, 1992). Monaghan (1986) found that students conceived of a variety of different kinds of numbers—'proper' numbers, such as whole numbers and familiar fractions, and 'improper numbers' such as infinite decimals, which 'go on forever'.

The shift from informal computational mathematics to formal axiomatic mathematics poses a new problem to the biological brain. Since the brain already has knowledge of many concepts, a definition (in a dictionary sense) is simply a way of identifying an *already existing concept*. The idea of *defining* a concept and then constructing its properties is quiet foreign. For example, when students have mental images of number lines with infinitesimal quantities on them, it is not easy for them to accept the axiom of completeness as it is contrary to the experiences of their biological brain (Li & Tall, 1993). There is another major shift in moving to axiomatic definitions and logical deductions to produce theorems. This involves not only the *process* of proof, but the *concept* of theorem, in which the (consequences of) theorems can be considered as entities that are logically manipulable to produce further theorems in a fully axiomatic theory. It has a structure which is quite different from their earlier experiences with numerical calculation and symbol manipulation. It shares many ideas in common with Euclidean proof (definitions, statements of theorems and deduction of consequences building a systematic theory). But it reveals a significant change from the developments of arithmetic, algebra and symbolic calculus. (See figure 2, based on Tall *et al.*, 2000.)

The development of symbol sense throughout the curriculum therefore faces a number of major re-constructions which cause increasing difficulties to more and more students as they are faced with successive new ideas that require new coping mechanisms. For many it leads to the satisfying

Formal definitions and proof	*defined* properties (axioms and definitions) *logical* processes (deductions) *formally constructed* concepts (theorems)
calculus	*computational* processes (rules) *manipulable* concepts (formulas)
(dynamic limit concept)	*potentially infinite* processes *arbitrarily small, close, or large* concepts (variable quantities)
(implicit algebraic concepts)	*implicit* processes (generalised properties) *manipulable* concepts (generalised expressions)
Algebra	*potential* processes (evaluating expressions) *manipulable* concepts (expressions)
Arithmetic	*computational* processes (arithmetic) *computational* concepts (numbers)

Figure 2 : changing meaning of symbols in arithmetic, algebra, calculus, and formal proof

immediate short-term needs of passing examinations by rote-learning procedures. The students may therefore satisfy the requirements of the current course and the teacher of the course is seen to be successful. However, if the long-term development of rich cognitive units is not set in motion, short-term success may only lead to increasing cognitive load and potential long-term failure.

COMPUTER ENVIRONMENTS FOR COGNITIVE DEVELOPMENT

Generic organisers

Given the many difficult transitions in the development of symbols and the consequent likelihood of increasing procedural thinking, we should consider ways in which we can enhance the students thinking processes to give richer cognitive units. The computer is powerful here because it can carry out any algorithms quickly and efficiently and represent the final result in a range of different representations. For example, the results may be represented *visually* and manipulated *physically* using a mouse to enable the student to build up embodied relationships that are part of a wider richer conceptual structure. This led me to design what I term 'generic organisers'.

- a *generic organiser* is an environment (or microworld) which enables the learner to manipulate *examples* and (if possible) *non-examples* of a specific mathematical concept or a related system of concepts. (Tall, 1989).

Generic organisers may be computer programs that give immediate feedback to the users explorations. They may also be physical objects such as Dienes' Blocks used in primary schools for exploring the concept of place value in different bases. Usually (but not always) they have visual and physical aspects that link to the fundamental workings of the human brain, its sensory inputs and subsequent actions.

An example which proved most valuable in making sense of the notion of variable as both process of evaluation and concept of expression is given in Tall and Thomas (1991). Realising that children were failing to give appropriate meaning to algebraic expressions, we introduced a physical game using a 'cardboard computer' to represent the storage of variables. The cardboard computer consisted of two large pieces of card, one being the 'screen' and the other being for 'storage', the latter having a number of small boxes on it for storing values. When an instruction was placed on the 'screen' the children acting as operators carry out the desired operation, storing and calculating expressions. For instance, if the command "A=1" is issued, the operators look for a box marked with the letter A—if not, they mark such a box by placing a small piece of card with the letter 'A' beside it—and then place a piece of card with the number '1' inside the box. If the next instruction is 'B=A+2', they set up a new store B, look inside store A, add two to the value, and place the result '3' in store B. The command 'PRINT B+1' would then cause the operators to look in store B, add on 1, then put the resulting number 4 on the 'screen'. This physical game concentrates on the students carrying out the *processes* of storage and evaluation. It can be used to illustrate equivalent expressions, by using, for instance, by inputting various values of A and then issuing instructions PRINT 2*(A+1), PRINT 2*A+2, to find that this always gives the same answer. The physical game is then linked to programming variables in BASIC, using programs such as INPUT A: PRINT 2*(+1):PRINT 2*A+2, to see that the two different looking expressions which represent different procedures of calculation always output the same value. Notice that the 'sense' of what is going on here focuses not on the actual procedures of calculation (the computer now does this internally) but on the *results* of the computation. The results are the same even though the symbolism and the procedures are different. In this way, the brain can focus on the important aspect of the situation (that two expressions always give the same results) in a way which is consonant with the manner in which the brain operates. It focuses on the important structure (provided always that the student actually *sees* this aspect of the structure) and helps the brain recognise that different expressions can be equivalent in the sense that they always give the same result. The computer *complements* brain activity by carrying out the operations internally and presenting the results so that the brain can focus on the displayed idea.

Cognitive roots

Designing generic organisers requires the selection of an important foundational idea to focus on. However, this idea is *not* a mathematical foundation for the theory. If it were, then the notion of *limit* would be used as a foundation for the study of calculus and we have shown earlier that the limit procept is seen by many students in terms of a potentially infinite *process* or as a *variable quantity* that is ‘abitrarily close’ or ‘infinitessimally small’.

Instead the introduction to the calculus requires a concept in the mathematical mind that is embodied in the biological brain. For both initial and also long-term success, a starting point is required which is both familiar to the student and also enables the student who wishes to take a deeper look into the theory to develop ‘appropriate’ intuitions. With this in mind I formulated:

- the notion of *cognitive root* (Tall,1989) as a cognitive unit which is (potentially) meaningful to the student at the time, yet contain the seeds of cognitive expansion to formal definitions and later theoretical development.

Cognitive roots for the calculus are simply *the notion of local straightness* (for rate of change/differentiation) and *area under the graph* (for cumulative growth/integration). The notion of local straightness was taken as the foundation of a generic organizer called *Magnify*, which allows the user to home in on a graph and draw a magnified portion in a second window. Once the concept of “seeing” the changing gradient of a locally straight curve is met, it is possible to have a second generic organiser which steps along the curve and plots the changing gradient.

The software can be even more powerful when designed as an environment for student exploration, teacher demonstration, and reflective discussion. Figure 3 shows a graph in the main window and a magnification of part of it in a second window. Here the graph in the second window still looks curved, but a further magnification will see it beginning to ‘look straight’.

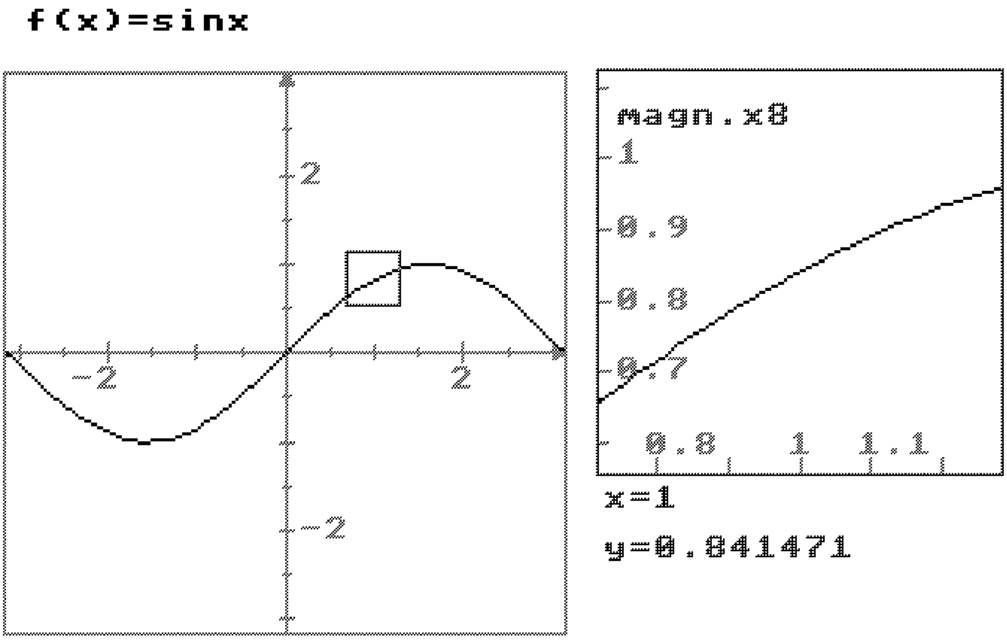

Figure 3 : Magnifying a (locally straight) graph to see how steep it is

If the software allows the user to step along the curve in the first window, redrawing the second window each time, then the user can *feel* and *see* the changing gradient along the curve. The notion of ‘local straightness’ is a cognitive root because it allows the gradient *function* to be seen as the changing gradient *of the graph itself*.

Lakoff and Johnson (1999) contend that *all* human thought is embodied, that is it is ultimately based on the activity of the human brain whose evolutionary design builds on bodily sensations and physical activity. This use of ‘local straightness’ is a deeply *embodied* concept. It is in tune with the basic functions of human perception and action. A vivid illustration of the embodied basis for thought arose in an episode in a calculus course where the teacher asked his students to use their knowledge of the derivative to determine the local maxima and minima of a given function. He drew a curve on the board with a local maximum and minimum, tracing along it with his finger rising, falling, then rising again. One of the students waved his hand up and down, tracing the shape of the curve passing over the maximum and said that the gradient would be positive before and negative after (Tall, 1986). This conception of a maximum is therefore more than just a learned response, it is an embodied conception that is based on the individual’s fundamental being.

I see the notion of local straightness to be more important than just illustrate the generic idea of pictorial gradient at times when it happened to work. To refine the meaning, *non-examples* were given of graphs which have corners, or are very wrinkled that they never look straight, providing anchoring concepts for non-differentiability (figure 4).

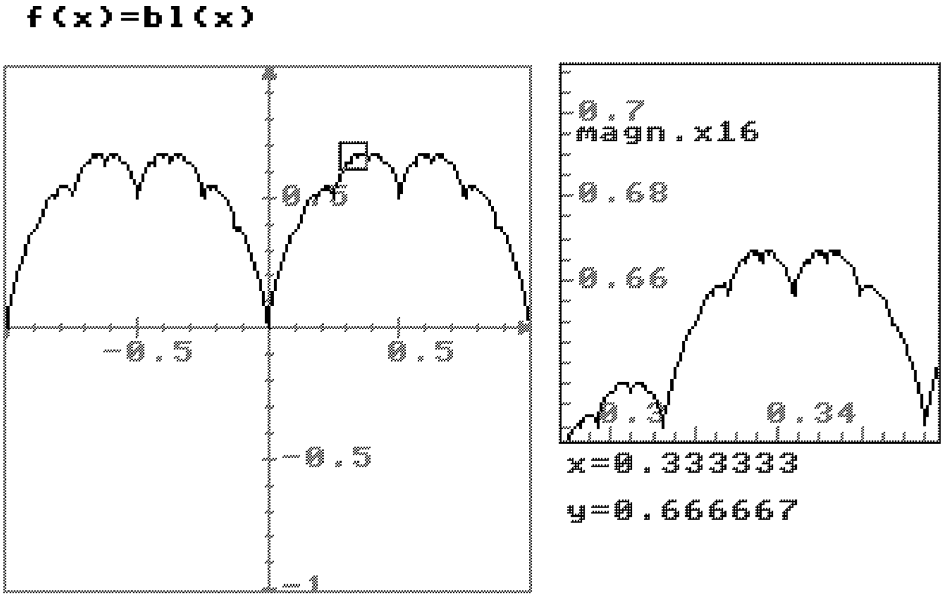

Figure 4: A graph which nowhere looks straight

Local straightness is therefore a profound embodied representation of the notion of gradient of a graph which appeals directly to the sense of vision and the mental images of curves. It can also be used to give insights into highly subtle ideas. For instance, if $f(x)$ is any differentiable function and $n(x) = 1000bl(x)/1000$ is a tiny wrinkled blancmange, and $f(x) + n(x)$ is *nowhere* differentiable (highly magnified it reveals the tiny wrinkles). Thus two graphs $f(x)$, $f(x) + n(x)$ *look identical* at a standard scale, but one is differentiable everywhere and the other nowhere (figure 5). This begins to move us from mere visualisation to more formal reflection. You cannot *see* the gradient, unless it is assumed that the picture is a faithful representation of the changing gradient with no tiny hidden wrinkles. This reveals the necessity of moving on from the visual basis of ideas to a more exact description of the functions in a precise sense.

This shows that the generic organiser *Magnify* can be used not only to get an informal idea of visual gradient, but also, with a little imagination, can demonstrate that the visual perception needs to be supported by something more substantial to develop a logically watertight theory. More

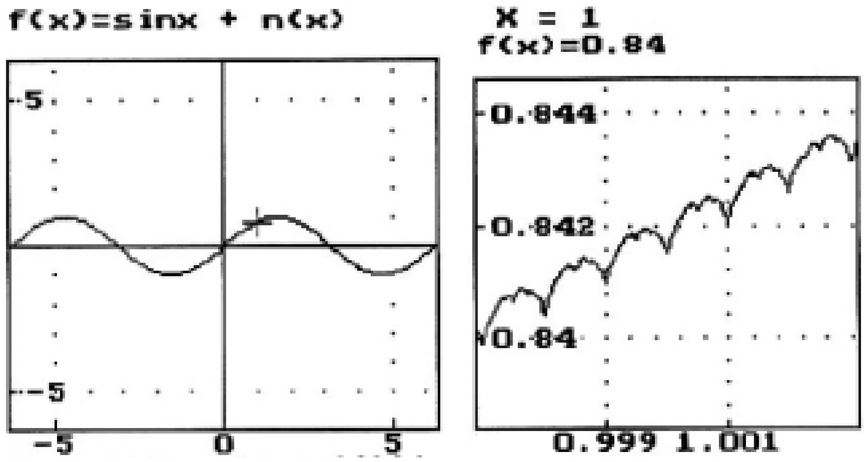

Figure 5: A familiar smooth-looking graph which magnifies to look 'rough' (Tall *et al* 1990)

generally, I expect all generic organisers to 'contain the seeds of their own destruction' in the sense that they are sufficiently sophisticated to show the limitations of their modelling process and the need for a fuller theoretical approach.

Embodied Local Straightness and Mathematical Local Linearity

The embodied notion of 'local straightness' is quite different from'local linearity' as used in most reform college calculus books. 'Local straightness' is a primitive human perception of the visual aspects of a graph. It has global implications as the individual looks *along* the graph and sees the changes in gradient, so that the gradient of the whole graph is seen *as a global entity*.

Local linearity, on the other hand, focuses more on what happens *at a single point* on the graph, having a linear *function* approximating the graph at that point. It is only when the concept of (symbolic) linear approximation is encapsulated to focus on its gradient, that the student is allowed to vary the point to give a global gradient function. It is a *mathematical* formulation of gradient, taken first as a limit at a point *x*, and only then varying *x* to get the formal derivative. Local straightness *remains at an embodied level* and links readily to the global view.

I see many mathematicians confusing these two quite distinct ideas, one intuitive and insightful, that can be used to 'see' highly subtle theoretical ideas in a meaningful embodied sense, the other formal and mathematical. We mathematicians with our mathematical minds full of logical formalisms are sometimes blind to the simple embodied realities that appeal to the biological brain and are capable of giving insights that can later underpin formal theory in the mathematical mind.

As an example, consider the inverse problem to that of differentiation. (No, this is *not* integration!) The problem is this—if I know the gradient of a function at any point, how can I build up the graph that has that gradient? In traditional calculus this is given in terms of linear differential equations in the form

$$\frac{dy}{dx} = F(x, y).$$

In traditional symbolic calculus this is attacked by a rag-bag of specific techniques suitable for a small number of types of differential equation. The meaning is (usually) lost. But the embodied meaning is plain. It is this: If I point my finger at any point (x,y) in the plane, then I can calculate the gradient of the solution curve at that point as $m = F(x,y)$ and draw a short line segment of gradient m through the point (x,y). This is a perfect opportunity to design a generic organiser on the computer. Simply write a piece of software so that when the mouse points at a point in the plane, a short line segment of the appropriate gradient is drawn, and as the mouse moves, the line segment moves, changing its gradient as it goes. As the solution curve is locally straight (because it has a gradient!) this line segment is *part* of the solution (at least, it *approximates* to part of a solution). The software allows the segment to be left in position by clicking the mouse. Hence by

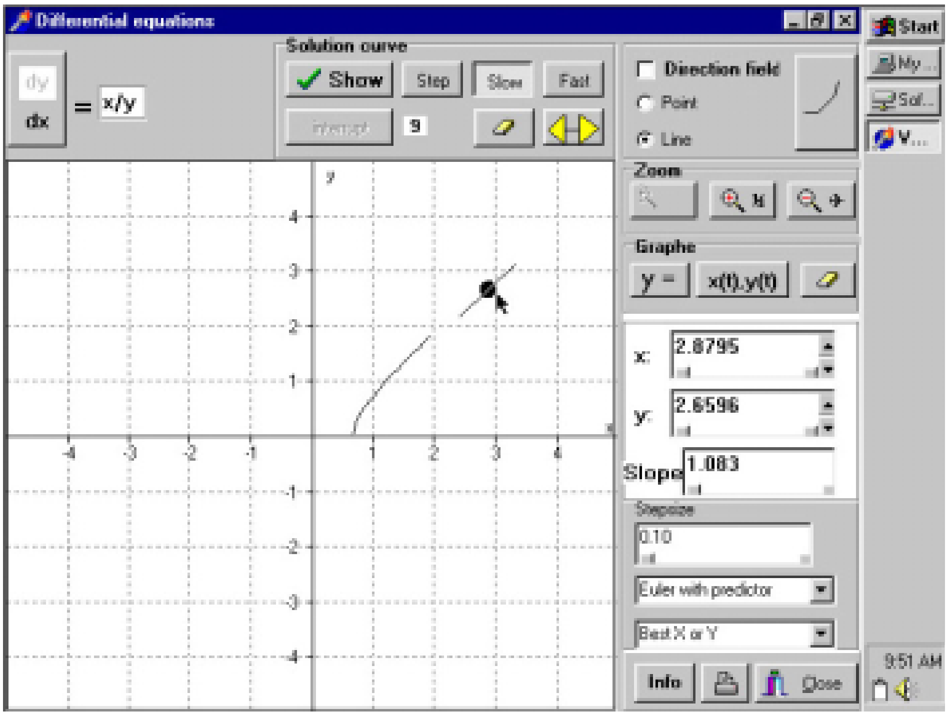

Figure 6: A solution of a first order differential equation built by hand, supported by software (using Blokland *et al*, (2000)).

pointing and clicking, then moving the line segment until it fits with the end of the curve drawn so far, an approximate solution curve can be constructed by sight and hand-movement—an embodied link between a first order differential equation and its solution. (See Figure 6.)

Continuity

In Tall, (1985), I showed how the notion of continuty can be illustrated for a real function All that is required is to stretch the graph much more horizontally than vertically. In figure 7 we see the

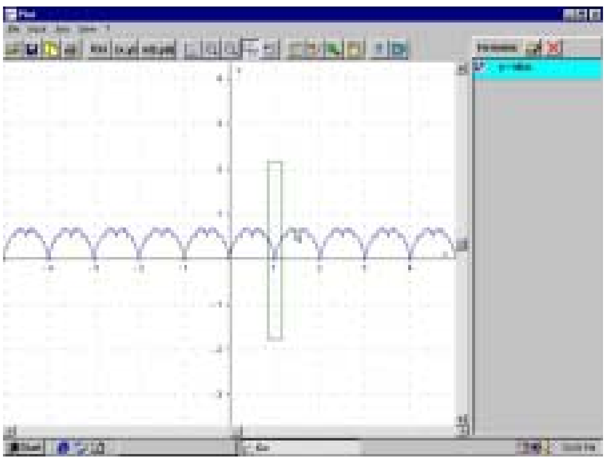

Figure 7: The blancmange graph and a rectangle to be stretched to fill the screen

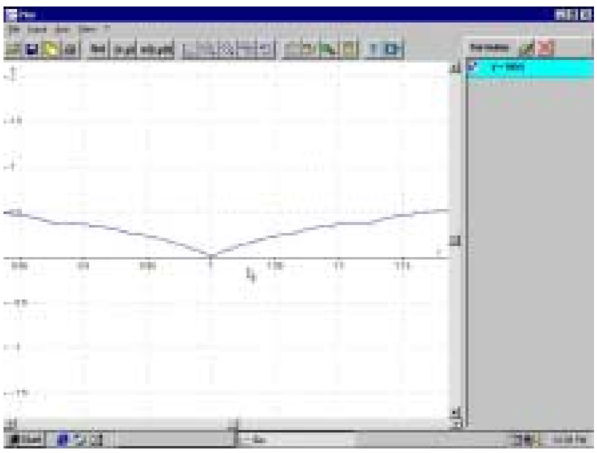

Figure 8: The blancmange function being stretched horizontally.

blancmange function with a rectangle that is tall and thin. This is stretched to give the picture in figure 8. It can be seen that the graph ‘pulls flat’ and that further stretching will flatten it horizontally. The translation from this embodied notion of continuity to the formal definition is not very far. Imagine the graph is drawn in a window with (x_0, $f(x_0)$) in the centre of the picture, in the centre of a pixel height 2ε. Suppose it ‘pulls flat”. Then the graph lies in a horizontal row of pixels and if the window is now of width 2δ, we have:

$|x - x_0| < \delta$ implies $|f(x) - f(x_0)| < \varepsilon$ [QED].

Think what happens when a very thin strip of area under a curve of width h is stretched horizontally. It gives a rectangle width h, height $f(x)$ (figure 9.) This can be used to show that the derivative of the area is $f(x)$, giving a pictorial version of the Fundamental Theorem. (Think about it!)

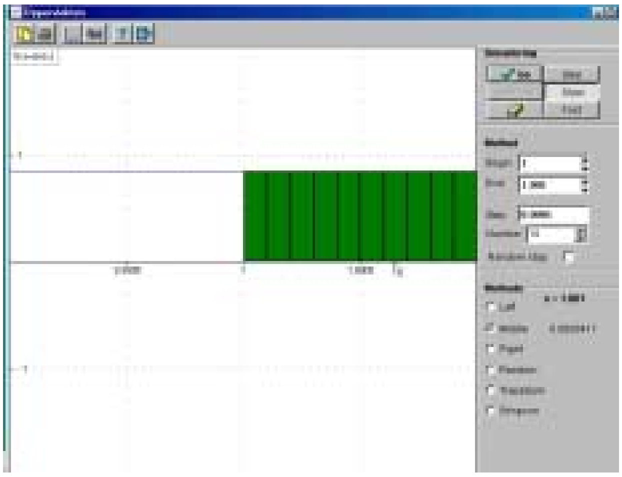

Figure 9: Area under sin*x* from 1 to 1.001 stretched horizontally

Embodied area and formal Riemann and Lebesgue integration

Just as the cognitive root of ‘local straightness’can be used to lead to more sophisticated theory, so the embodied notions of ‘area’ and ‘area-so-far’ can support Riemann and even Lebesgue integration. The use of technology to draw strips under graphs and calculate the numerical area is widely used. With a little imagination, and well-planned software, it can be used to give insight into such things as the sign of the area (taking positive and negative steps as well as positive and negative ordinates) and to consider ideas such as how the notion of continuity relates to the notion of integration. For these ideas, I refer the reader to selected papers such as: Tall (1985, 1991a, 1992, 1993, 1995, 1997), These may be downloaded from my web-site:
http://www.warwick.ac.uk/staff/David.Tall

I conclude this presentation by showing a few visual examples of various sophisticated concepts in mathematical analysis.

The blancmange function is continuous (Tall, 1982), and therefore its area function is differentiable. Figure10 shows the numerical area function for the blancmange and the *gradient of the area function*. This looks like the original graph. Of course it does, because the derivative of the area is the original function again.

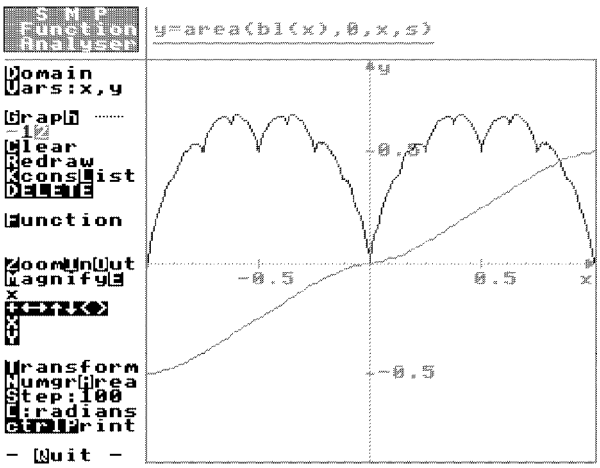

Figure 10: the area function of the blancmange and the derivative of this area (from Tall, 1991b)

Another much more interesting situation is to consider the ‘area’ under a function which has a number of discontinuities. The function x–int(x) is discontinuous at each integer and is continuous everywhere else. The area function is continuous everywhere and is also differentiable everywhere that the original function is continuous (figure 11). However, at the integer points, if the graph of the area function is magnified, it can be seen to have a corner at each integer point, because here the area graph has different left and right gradients (figure 12). If you look at the change in the area under the function you may be able to *see* why this happens.

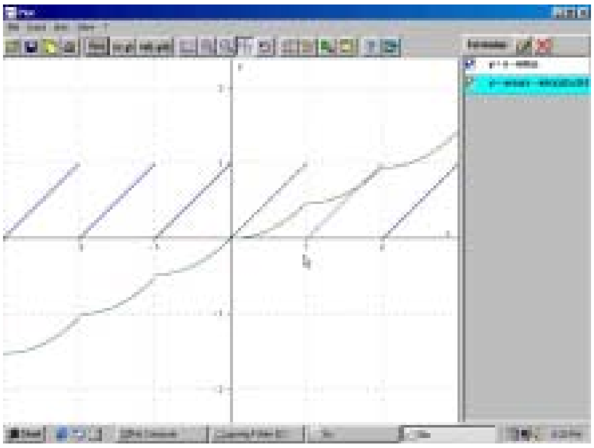

Figure 11: the area function for *x–int*(*x*)

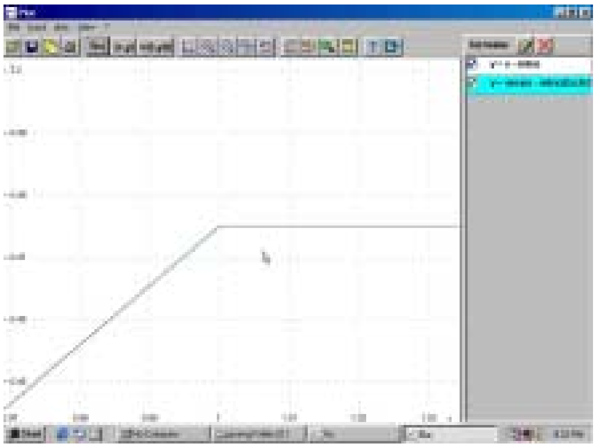

Figure 12: The area function magnified

It was an ambition of mine to draw functions such as *f(x)=x for x rational*, *f(x)=1–x for x irrational.* The fact that this was impossible for numerical calculations on a computer (which are all rational) did not deter me. In Tall (1993), I found a method that enabled me to make such a model. For any number *x*, I simply calculated the sequence of rational approximations to *x* using the method of continued fractions. This behaves differently for rational *x* (where the sequence ends by equalling *x*) and irrational *x* (where the denominators grow without limit). Modelling what happened to this sequence allowed me to subdivide numbers into two disjoint sets numerically, which I called ‘pseudo-rational’ and ‘pseudo-irrational’. I also programmed a routine plotting random points, which were mainly ‘pseudo irrational’ and a second routine that plotted mainly ‘pseudo-rationals’.

Figure 12 shows pictures of the function which is *x* on the rationals and 1–*x* on irrationals together with a graph for the area ‘under the graph’ from 0 to *x*. This uses the mid-ordinate rule with a fixed with (rational) step. It encounters mainly (pseudo-) rationals where *f* (*x*)=*x*, so the resulting area function approximates to $x^2/2$. (figure 13). When the area is calculated using a random step-length and a random point in the strip to calculate the area, it encounters mainly (pseudo-) irrationals where the function has values *f*(*x*) = 1–*x*. The area function drawn in this case reflects the latter formula (figure 14). (Here I have drawn several plots of the area curve. Because of the errors calculating pseudo-rationals and irrationals, there are small discrepancies with the random area that is slightly different each time.)

I used this software to discuss the area under such graphs (Tall, 1993). Students who were not mathematics majors and who would normally not cope very well in an analysis course were able to discuss this example intelligently, noting that ‘a random decimal is highly unlikely to repeat, so random decimals are almost certainly irrational’. This led to a highly interesting discussion on the ‘area’ under ‘peculiar’ graphs which began to move the thinking on from Riemann integration to Lebesgue integration. It was only a glimpse of the ideas for these students, but it was a glimpse that they could empathise with. It shows how the mathematical mind can gain insights from visuo-spatial ideas in areas where the formal theory would be far too abstruse. But, for some of those who later do go on to the formal theory, visualization can provide a powerful cognitive foundation.

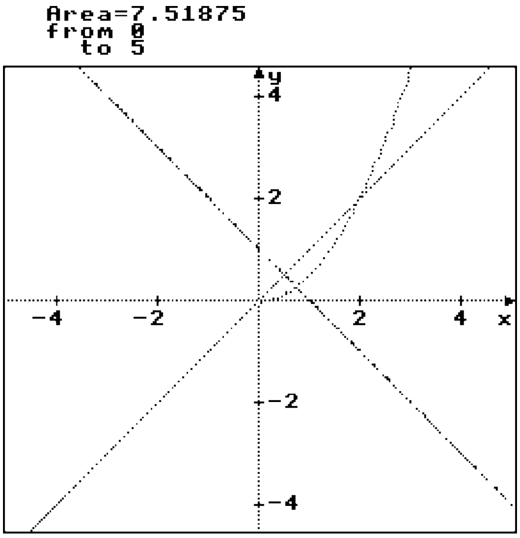

Figure 12: The (pseudo-) rational area

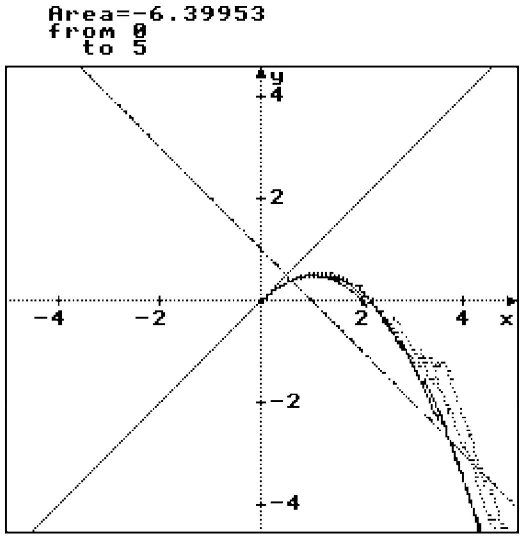

Figure 13: The (pseudo-) irrational area

Epilogue

In this presentation I have shown how the human mind does not always do mathematics logically, but is guided by a concept image that can be both helpful and also deceptive. It may seem that symbolism is more precise and safe than visualisation, but I produced evidence to show how the cognitive development of symbols in arithmetic, algebra and calculus have many potential cognitive pitfalls. In arithmetic, algebra and calculus I showed how a combination of visual and enactive experiences can complement symbolic methods. To do this requires a carefully prepared curriculum and the guidance of a teacher as mentor to focus on ideas that are fundamental and generative. I reported how the use of local straightness and visual ideas of area can be cognitive roots that are foundational in building an embodied understanding of the calculus, taking the ideas to a stage where, given careful guidance, ideas can be motivated that are part of the formal theory of differentiation, continuity and integration. To do this requires more than mathematics and more than a knowledge of cognitive growth. It requires a special approach to mathematical thinking that supports the concept imagery of the biological brain by interaction with a computational computer to produce a versatile mathematical mind.

References

Anderson, C. (1997). Persistent errors in indices: a cognitive perspective, Ph.D. Thesis, University of New England, Armidale, Australia.

Bakar, M & Tall, D.O., (1992). Students' Mental Prototypes for Functions and Graphs, *International Journal for Mathematics Education in Science and Technology*, **23** (1), 39-50.

Barnard, A. D.. & Tall, D. O. (1997). Cognitive Units, Connections, and Mathematical Proof. In E. Pehkonen, (Ed.), *Proceedings of the 21st Annual Conference for the Psychology of Mathematics Education,* Vol. 2 (pp. 41–48). Lahti, Finland.

Blokland, P., Giessen, C, & Tall, D. O. (2000). *Graphic Calculus for Windows*. (see www.warwick.ac.uk/staff/David.Tall.)

Bruner, J. S. (1966). *Towards a Theory of Instruction*. Cambridge: Harvard.

Butterworth, B. (1999). *The Mathematical Brain*, London: MacMillan.

Carter, R. (1998). *Mapping the Mind*. London: Weidenfeld & Nicholson.

Cornu, B. (1991). Limits. In D. O. Tall (Ed.). *Advanced Mathematical Thinking* (pp. 253–166). Dordrecht: Kluwer.

Crick, F. (1994). *The Astonishing Hypothesis*, London: Simon & Schuster.

Crowley, L. (2000). *Cognitive Structures in College Algebra*, Ph.D. thesis, University of Warwick.

Crowley, L. & Tall, D. O. (1999). The Roles of Cognitive Units, Connections and Procedures in achieving Goals in College Algebra. In O. Zaslavsky (Ed.), *Proceedings of the 23rd Conference of PME, Haifa, Israel*, 2, 225–232.

Davis, R. B., Jockusch, E., McKnight, C. (78). Cognitive processes in learning algebra, *Journal of Children's Mathematical Behavior*, 2 (1).

Dehaene, S. (1997). *The Number Sense*. New York: Oxford University Press.

Devlin. K. (2000), *The Maths Gene: Why everyone has it, but most people don't use it*, London: Weidenfeld & Nicholson.

Edelman, G. M. (1992). *Bright Air, Brilliant Fire*, NY: Basic Books, reprinted Penguin 1994.

Edelman, G. M. & Tononi, G. (2000). *Consciousness: How Matter Becomes Imagination*. New York: Basic Books.

Gray, E. M. & Tall, D. O. (1994). Duality, ambiguity and flexibility: a proceptual view of simple arithmetic, *Journal for Research in Mathematics Education*, **26**, 115–141.

Greenfield, S. (1997). *The Human Brain: A Guided Tour*. London: Weidenfeld & Nicholson.

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.

Lakoff, G. and Johnson, M. (1999). *Philosophy in the Flesh*. New York: Basic Books.

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*. 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.

Pinker, S. (1997). *How the Mind Works*. New York: Norton.

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. (1986). *Building and Testing a Cognitive Approach to the Calculus using Interactive Computer Graphics*, Ph.D. Thesis, University of Warwick, UK.

Tall, D. O. (1989). Concept Images, Generic Organizers, Computers & Curriculum Change, *For the Learning of Mathematics*, **9**,3, 37–42.

Tall, D. O. (1991a). Recent developments in the use of the computer to visualize and symbolize calculus concepts, *The Laboratory Approach to Teaching Calculus*, M.A.A. Notes Vol. 20, 15–25.

Tall, D. O., (1991b). *Real Functions and Graphs*, (software for BBC compatible computers), Cambridge University Press: Cambridge, UK.

Tall, D. O. (1992). Visualizing differentials in two and three dimensions, *Teaching Mathematics and its Applications*, **11** 1, 1–7.

Tall, D. O. (1993). Real Mathematics, Rational Computers and Complex People. In Lum, L. (ed.), *Proceedings of the Fifth Annual International Conference on Technology in College Mathematics Teaching*, 243–258.

Tall, D. O. (1997) Functions and Calculus. In A. J. Bishop et al (Eds.), *International Handbook of Mathematics Education*, 289–325, Dordrecht: Kluwer.

Tall, D. O., Blokland, P. & Kok, D. (1990), *A Graphical Approach to the Calculus*, (software for IBM compatible computers), Sunburst Communications Inc, Pleasantville, NY.

Tall, D. O, Gray, E M., Ali, M. B., Crowley, L., DeMarois, P., McGowen, M., Pitta, D., Pinto, M., Thomas, M., &d Yusof, Y. (2000). Symbols and the Bifurcation between Procedural and Conceptual Thinking, *Canadian Journal of Science, Mathematics and Technology Education* 1, 80–104.

Tall, D. O. & Thomas, M. O. J. (1991). Encouraging Versatile Thinking in Algebra using the Computer, *Educational Studies in Mathematics*, **22** 2, 125–147.

Tall, D. O. & Vinner, S. (1981). Concept image and concept definition in mathematics, with special reference to limits and continuity, *Educational Studies in Mathematics*, **12**, 151–169.

Vinner, S. (1983). Concept definition, concept image and the notion of function, *The International Journal of Mathematical Education in Science and Technology* **14**, 293–305.

Williams, S. R. (1991). Models of limit held by college calculus students, *Journal for Research in Mathematics Education*, 22 (3), 237–251.

Wood, N. G. (1992). *Mathematical Analysis: a comparison of student development and historical development*. Ph.D. thesis, Cambridge University, UK.