Using Computers Environments to Conceptualize Mathematical Ideas[†]

David Tall

Mathematics Education Research Centre,Science Education Department,
University of Warwick, COVENTRY CV4 7AL, United Kingdom

ABSTRACT

In addition to the computer providing us with new tools to *use* in mathematics, it also provides opportunities to develop flexible software to help students (and teachers) to *understand* fundamental mathematical concepts. The “Graphic Approach to the Calculus”, designed by the author, is an example of such an environment, in which computer software is used for teachers to introduce – and students to explore – fundamental ideas such as the notion of a derivative, the meaning of the integral as a sum, the idea of a solution of a differential equation. More recently the ideas have been further developed in the new School Mathematics 16–19 Curriculum using computer graphics to visualize ideas about functions, graphs and the calculus. This article outlines the theory underlying the use of such pieces of software – which I term *generic organizers* because they are designed to encourage the user to organize their conceptions from typical (generic) examples – and briefly reviews the basic ideas underlying several of the programs. The ideas will be framed within a broader theory of *versatile learning*, in which the usual sequential activities of carrying out calculations and deductions in mathematics are complemented by global overviews of how the ideas link together – providing a powerful way in which computer technology may be used to enhance human thought processes.

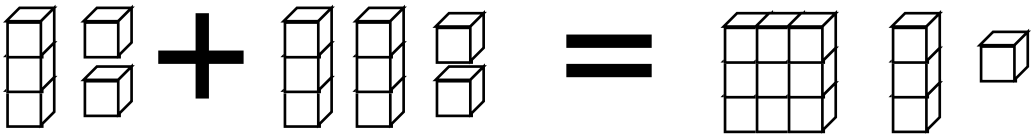
1. Human thought processes and the use of the computer

Human beings are perverse and fascinating creatures who are distinguished from the rest of the animal kingdom by their seemingly unbounded capacity to invent and use *tools* to extend their limited physical capabilities. The computer is one such tool – potentially one of the most powerful. Its mode of operation is somewhat different from the human mind, and precisely because of this, it can be used to *supplement* and *complement* human thinking processes.

Tools have been used for conceptual purposes in mathematics for centuries. An *abacus* can be used to perform complex arithmetic calculations through routinized physical manipulations. More recently, physical tools have been developed to conceptualize mathematical ideas and foster understanding. For instance, Dienes blocks are designed to give insight into the notion of place value in arithmetic in different bases (figure 1).

[†] This article is based on the theoretical developments and computer programs developed by the author in publications 4-13 in the list of references. The programs in references 5-8, 11,12 are for BBC, Nimbus and Archimedes computers, those in reference 13 for IBM compatibles.

This paper appeared in *Proceedings of Conference on New Technological Tools in Education*, Nee Ann
Polytechnic, Singapore, 55-75, 1990. It is based on earlier publications of the author (see references).



12 + 22 in base 3, regrouped to give 111

figure 1

Here the child carries out the manipulations, regrouping the blocks to give the result, whose correctness he must judge for himself. Computer environments have the additional advantage that they can be programmed to carry out complex manipulations and provide feedback for the learner†.

My main aim in the development of computer software has been to design environments which allow the teacher and student to manipulate mathematical concepts and processes to gain greater insight into the ideas. The software is designed not only to give *examples* of the processes at work, but also the facility to look at a broader class of situations where the ideas *fail*, for it is my belief that many of the misconceptions in human learning arise from inadequate experiences that do not exhibit the full range of possibilities.

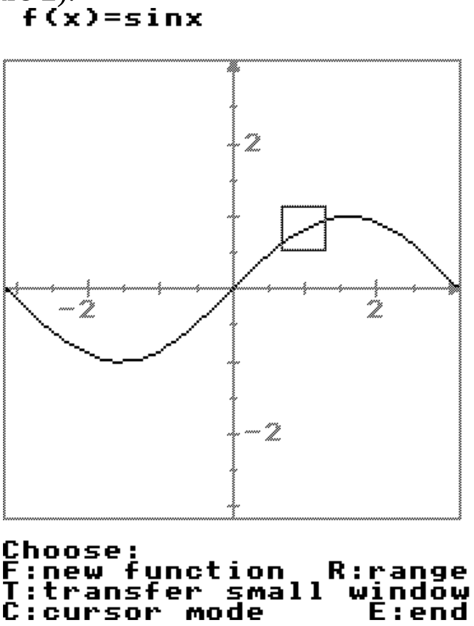
More advanced mathematics abounds with definitions that say explicitly what concepts *are*, but only implicitly indicate what the concepts *are not*. A continuous function is... A convergent sequence is ... A derivative is ... A Riemann integral exists if ... and so on. We attempt to teach students advanced mathematics by giving them definitions and making deductions – a process which is known to be unsuccessful with the majority of students.

Suppose instead that we give the students an environment in which a new concept can be explored – to see instances where it is true and also where it fails. For instance, what do we mean by the notion of a differentiable function? The traditional method – definitions of limits, symbolic and geometric ideas of limiting concepts – is known to cause significant conceptual problems for students. (For example, they invariably believe that the limiting process “gets arbitrarily close” but “cannot reach” the final position and are initially uncomfortable with the limiting process, only taking solace in the routine manipulation of the rules of differentiation which they see as the “safe” part of the process they can cope with.) A method with immediate appeal is to use a computer to magnify a small portion of a graph.

† A program for the Macintosh computer which continuously links together pictures of Dienes’ blocks being manipulated by the user and the numerical equivalents has been designed by Prof Pat Thompson of Illinois State University, USA.

2. A practical example of a computer environment in action

The **Magnify** program of Graphic Calculus[6,13] allows the user to draw a graph in one window on the computer screen and to magnify a selected portion of the graph in a second window (figure 2).



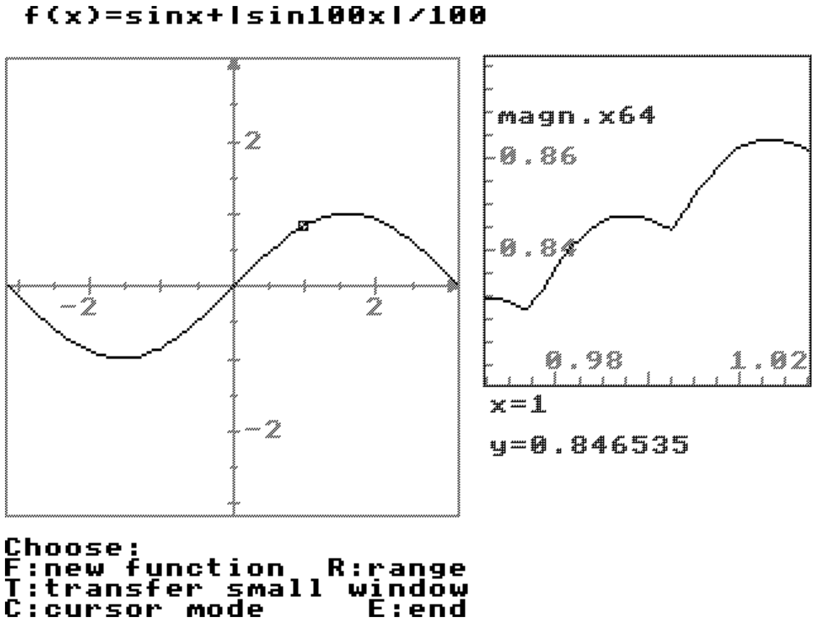
```
x=1
y=0.841471
```

figure 2

Used without guidance this will lead to pupils discovering various properties for themselves, for instance:

- as the graph is magnified more and more the picture gets less curved,
- a highly magnified portion of the graph looks (fairly) straight,
- the picture of the graph is not always as smooth might be expected.

Of these, the first is a highly perceptive insight that later will link with subtle ideas about curvature, the second is the fundamental idea (“local straightness”) underlying the notion of differentiability and the third is a red herring, brought about by the limitations of the computer picture made up of a finite number of pixels, or points of light. This will need teacher intervention, perhaps to participate in the discussion of the pixel limitations, but almost certainly to focus the students on graphs of functions which are *not* locally straight. Students may suggest ideas such as $f(x)=|x|$, or teacher guidance may be necessary. Once started it is possible to lead to more interesting examples, such as $|\sin x|$ with “corners” at every multiple of π, which magnifies at each of these points to look like two half-lines meeting at an angle, or more subtle cases, such as $\sin x+|\sin 100x|/100$, which looks like $\sin x$ to a normal magnification, but reveals corners when magnified (figure 3).

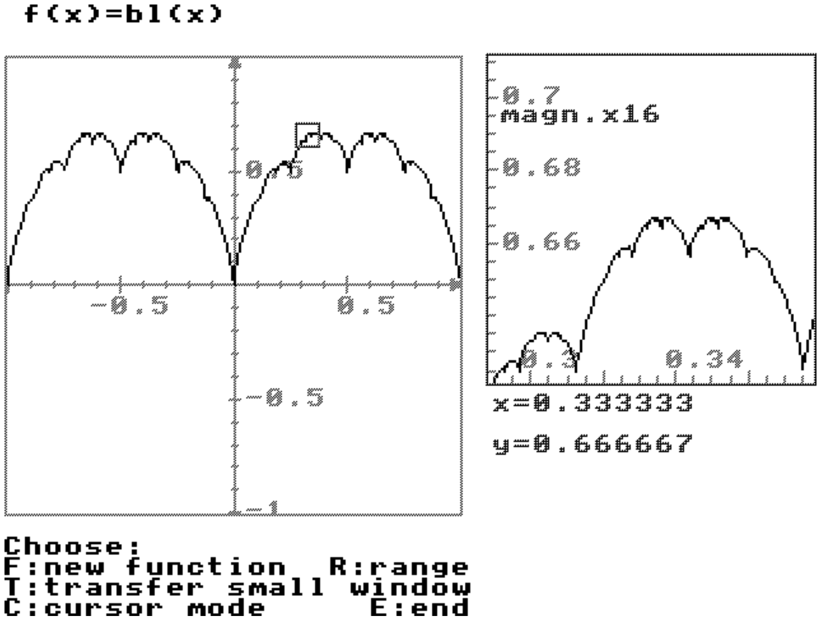


```
f(x)=sinx+|sin100x|/100
```

```
x=1
y=0.846535
Choose:
F:new function  R:range
T:transfer small window
C:cursor mode      E:end
```

figure 3

The software is designed to cope with much more subtle functions, for instance the blancmange function, f(x)=bl(x), which is a fractal and when highly magnified, smaller blancmanges are revealed growing everywhere (figure 4)†.



```
f(x)=bl(x)
```

```
x=0.333333
y=0.666667
Choose:
F:new function  R:range
T:transfer small window
C:cursor mode      E:end
```

figure 4

† This screen dump has been taken from the *Archimedes* version of the program. The same facility, with a different screen layout is available for IBM compatibles

It is so wrinkled that *nowhere* does it magnify to look straight. This is cognitively not a difficult concept, and it shows that computer technology can be used to model the notion of a *nowhere-differentiable* function in the very early stages of the calculus. I have demonstrated elsewhere on many occasions how this idea can be proved in a way which is cognitively satisfying to beginners yet contains all the ingredients necessary for a formal proof [4,6,13].

3. The growth of a theoretical perspective

An environment which enables the exploration of *examples* and (where possible) *non-examples* of a process, or concept, I term a **generic organizer**. The Magnify program is a generic organizer for the concept of "local straightness", which experience shows to be a good starting point for the theory of differentiability. Dienes blocks are an example of a generic organizer (for the concept of place value in arithmetic), but this organizer is *passive*. The user must operate on the blocks and must make the necessary judgements as to the theory underlying the manipulation of the blocks. The Magnify program is a *cybernetic* organizer. It not only accepts the manipulative acts of the user (giving instructions to draw a graph of a given function over a given range), it takes pre-programmed action to carry out the consequences of those acts (drawing graphs, magnifying selected portions of graphs). These activities help to focus the user on the notion of "local straightness", which is the underlying reason for the design of the program.

There are many software programs in education which are designed to judge whether a student's input is correct or not. It is a fundamental axiom of my development of generic organizers that the software makes no judgements. It simply returns to the user the results of the user's command and it is the user who must make a personal assessment of the correctness of the situation. Thus the generic organizer remains a *tool* and the human user remains *in control*, albeit in debt to the organizer's ability to process the calculations required by the user's decisions.

Note that the explicit fundamental idea underlying the Magnify program is that of *local straightness* rather than the *mathematical limit concept*. The local straightness idea has two fundamental and complementary properties:

- it is meaningful within the current cognitive development of the pupil,
- it is a powerful mathematical idea that can be used as a starting point for the theory.

The first of these has been shown through empirical research[9], the latter is a matter of re-thinking mathematical analysis in such a way that the formal theory can be seen to be grow from a formal translation of local straightness. Local straightness leads to the idea of differentiability and the converse – building up a locally straight curve knowing its derivative – is the fundamental idea of differential equations. At a higher level, local straightness is the one-dimensional intuitive version of the form notion of a differentiable manifold (a locally euclidean – or locally flat - topological space).

A fundamental idea with these two properties I call a *cognitive root.* It is an idea which can be planted in the soil of the human mind to find fertile links on which it can grow into a formal theory. A cognitive root is very different from a logical foundation, for the latter only satisfies the second principle : a starting point for a formal development. Experience of the "new math" has shown us that logical foundations may fail to be cognitive roots. The "new math" failed because it did not take root in the minds of the learners. The "new *computer* math" founded on *cognitive roots* has a greater chance of success because it is designed both to build from the student's current position and also to lead to the power of the formal mathematics.

The example of the Magnify program raise a number of pertinent points:

- Given an appropriate computer environment, students can discover fundamental mathematical ideas, based on their cognitive state at the time,

- Without teaching, they may not understand the full significance of their discoveries,

- Subtle and powerful ideas, particularly those related to the periphery of the concept and to non-examples, are almost certain to need specific teaching,

- A learning sequence using generic organizers may occur in a different order from a formal mathematical development.

Before the arrival of the computer it was a common belief that students learn best when they grow from their own experience. However, the introduction of the calculus shows that in the absence of the computer, the need to develop a rich enough collection of examples and non-examples was so onerous that the student would be swamped with the sheer variety and detail of those examples. At the beginning of this century, Poincaré[3] said:

> If logic were the sole guide of the teacher, it would be necessary to begin with the most general functions, that is to say with the most bizarre.

and he then went on to dismiss such a fantasy. Of course it was crazy then to attempt teach the calculus beginning with the bizarre notion of an everywhere non-differentiable function because the calculations necessary to exhibit such a function involved *explicit* references to subtle limiting processes. But the arrival of the Magnify program, which handles the calculations *implicitly* and *internally* – together with the existence of simple-to-explain functions like the blancmange function – mean that a wide variety of examples and non-examples can be explored in a single session with a generic organizer. Because the complexity of the detailed calculation is concealed within the workings of the computer program, teacher and students are freed from the computational burden to concentrate on the essential idea: is the graph of the function locally straight or not? Within this context it is possible to concentrate once and for all on the powerful idea from which the theory can blossom: *the cognitive root of local straightness*.

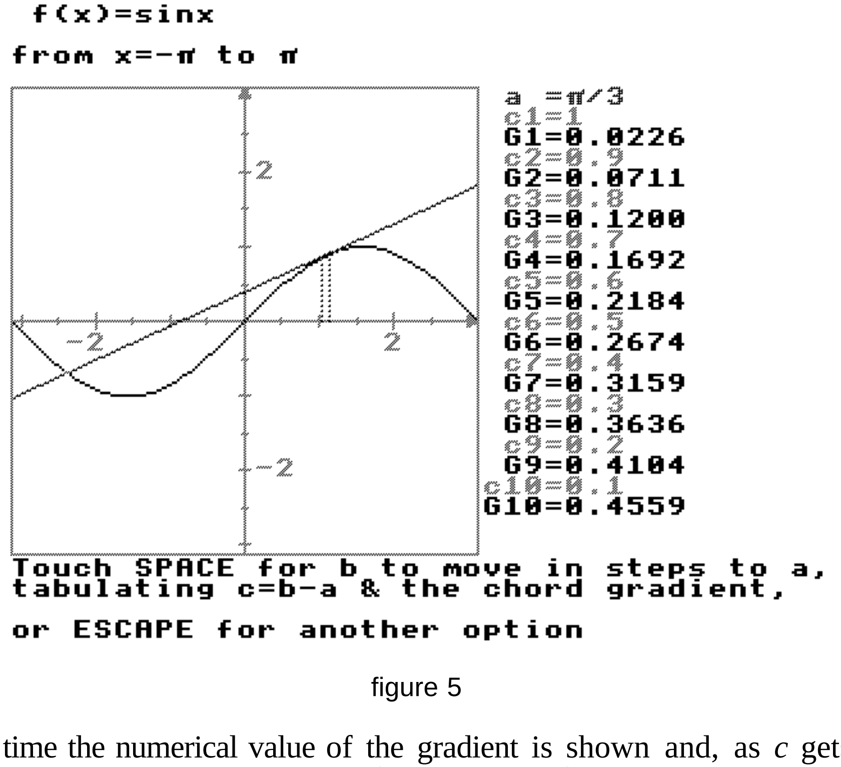
This leads to a new way of approaching the curriculum, in which learners can gain initial insight into the fundamental ideas at their current cognitive level, linking them together in a global, often visual way, in addition to the usual sequential mathematical processes which are the basis of the traditional curriculum. Generic organizers are designed to help the user gain a preliminary global/holistic insight to provide a context for relational understanding of logical/sequential processing which follows. The combination of these complementary modes of global/holistic and logical/sequential learning is termed *versatile*[1]. Research at Warwick University has shown that a more versatile approach to the learning of mathematics results in significant improvements in the ability of students to solve non-routine problems related to the ideas, without any diminution in their ability to perform traditional manipulations [1,9,14].

4. Examples of Generic Organizers in Graphs, Functions and the Calculus

4.1 Gradient†

The cognitive root of the calculus is the notion of local straightness. A student who understands this can look at the graph of a function and, in his or her mind's eye, be able to mentally zoom in and *see* the changing gradient of the curve (provided, of course, that the picture faithfully represents the situation and there are no smaller hidden wrinkles!).

The Gradient program has two different generic organizers – one which looks at the gradient of a chord through points (x,f(x)), (x+h,f(x+h)) for fixed x as h decreases in steps – the other fixes h and considers the changing gradient as x increases. The first of these mirrors the way that the symbolic or geometric limit is first introduced in traditional calculus. It has the advantage that the student can see the chord through two close points stabilizing as h gets small. (In figure 5, a=x, h=c and b=x+h).



```
f(x)=sinx
from x=-π to π

a =π/3
c1=1
G1=0.0226
c2=0.9
G2=0.0711
c3=0.8
G3=0.1200
c4=0.7
G4=0.1692
c5=0.6
G5=0.2184
c6=0.5
G6=0.2674
c7=0.4
G7=0.3159
c8=0.3
G8=0.3636
c9=0.2
G9=0.4104
c10=0.1
G10=0.4559

Touch SPACE for b to move in steps to a,
tabulating c=b-a & the chord gradient,

or ESCAPE for another option
```

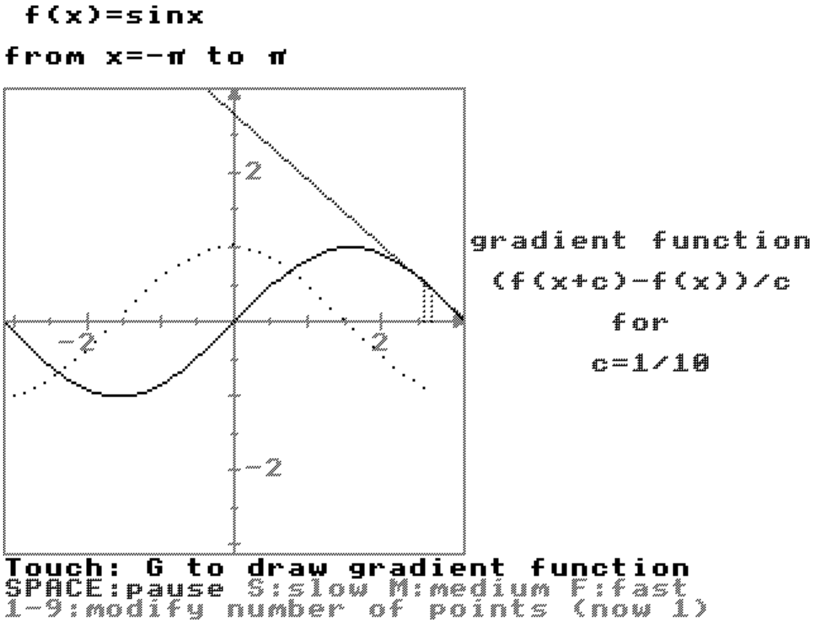
figure 5

At the same time the numerical value of the gradient is shown and, as c gets small, the gradient will be seen to be stabilizing at a fixed value. *Of course it will!* The student has a

† Available for all IBM compatibles, the BBC computer, Archimedes, Nimbus PC.

new understanding – in the mind’s eye the graph is locally straight, so provided that the points are close enough, they are on a virtually straight part of the graph and the gradient between any two points on the graph is virtually the same. The more the magnification is performed, the better the approximation to a straight line and the better the approximation of the gradient to a fixed value. If the student so desires, it is possible to zoom in close up to a point on the graph and see the graph as a straight line to give substance to the mental picture.

The second option is the one with the most powerful mental imagery as it encourages the learners to gain a *global* view of the gradient, moving dynamically along the curve to *see* the changing gradient (figure 6).



f(x)=sinx

from x=-π to π

gradient function

(f(x+c)-f(x))/c

for

c=1/10

Touch: G to draw gradient function

SPACE:pause S:slow M:medium F:fast

1-9:modify number of points (now 1)

figure 6

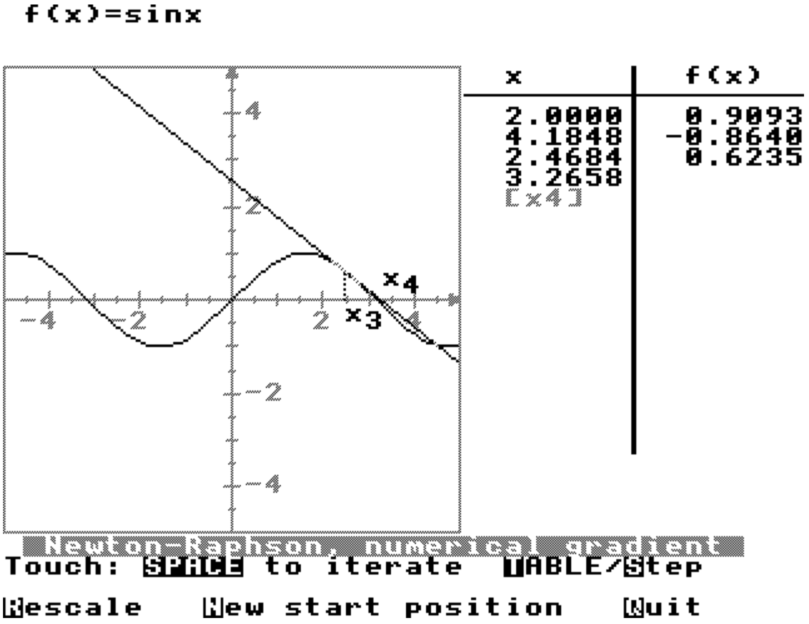
The seed is sown that the gradient changes as a *function* of the value of *x*...

4.3 Solutions of Equations[†]

A short stretch of a locally straight graph can be approximated by a straight line, giving rise to various methods of approximating to a solution, depending on how the straight line approximation is calculated. For instance, figure 7 calculates successive approximations x_1, x_2, x_3, x_4, ... to a root using a numerical version of the Newton-Raphson. Having found x_n, the line through the points on the curve with x coordinates x_n, x_n+h is used as an approximation the curve. Where this line meets the x-axis is the next approximation x_{n+1}. An autozoom feature allows the user to zoom in on the process – which will reveal the local straightness of the curve and why the method is effective.

[†] Available the BBC computer, Archimedes, Nimbus PC and, in a slight different form for IBM compatibles.

Opportunities for versatile learning are immediately available. In addition to studying the local theory of convergence (what happens when the process is near a root) it is also possible to use the software to investigate the more intricate (even catastrophic!) global theory of what happens when the process is begun in in various regions further away.



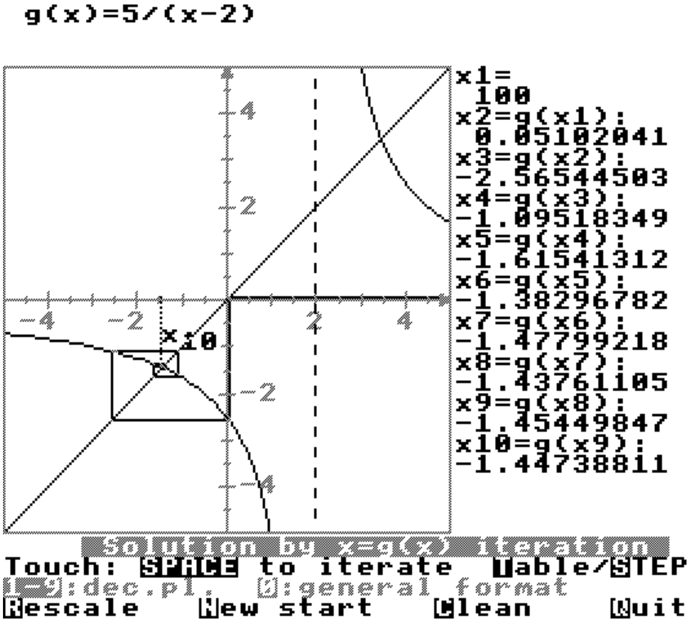
f(x)=sinx

```
x      | f(x)
2.0000 |  0.9093
4.1848 | -0.8640
2.4684 |  0.6235
3.2658 |
[x4]   |
```

4
2
-4
-2
2
-2
-4
x3
x4
Newton-Raphson, numerical gradient
Touch: SPACE to iterate TABLE/Step
Rescale New start position Quit

figure 7

Figure 8 shows the standard iteration process to solve *x*=g(*x*) by replacing *x* by g(*x*) .



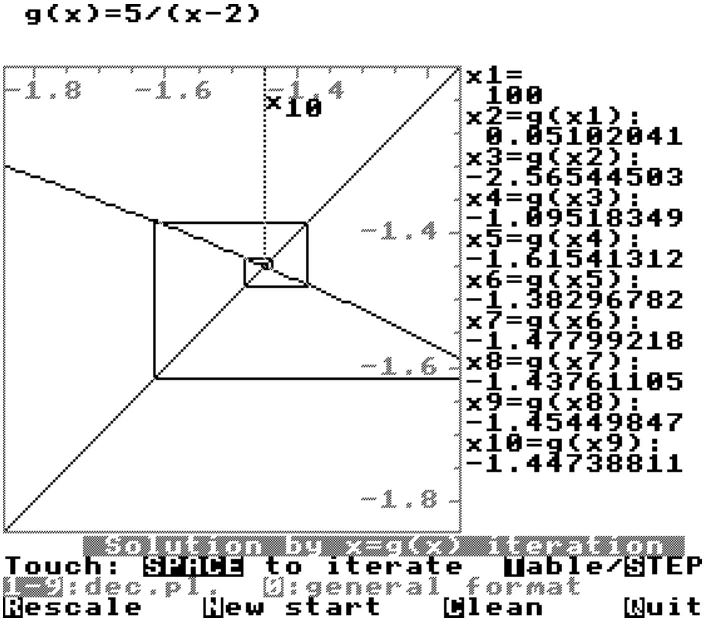
g(x)=5/(x-2)

```
x1=
 100
x2=g(x1):
 0.05102041
x3=g(x2):
-2.56544503
x4=g(x3):
-1.09518349
x5=g(x4):
-1.61541312
x6=g(x5):
-1.38296782
x7=g(x6):
-1.47799218
x8=g(x7):
-1.43761105
x9=g(x8):
-1.45449847
x10=g(x9):
-1.44738811
```

4
2
-4
-2
2
4
-2
-4
x10
Solution by x=g(x) iteration
Touch: SPACE to iterate Table/STEP
1-9:dec.pl. 0:general format
Rescale New start Clean Quit

figure 8

Sometimes the iteration process homes on a solution and sometimes it does not. But why? Rescaling the picture during operation may give insight. Once the magnification is high enough to represent a small part of the graph y=g(x) as almost straight, it becomes apparent that each iteration is just moving closer by the same proportional quantity (figure 9).



```
g(x)=5/(x-2)

x1=
 100
x2=g(x1):
 0.05102041
x3=g(x2):
-2.56544503
x4=g(x3):
-1.09518349
x5=g(x4):
-1.61541312
x6=g(x5):
-1.38296782
x7=g(x6):
-1.47799218
x8=g(x7):
-1.43761105
x9=g(x8):
-1.45449847
x10=g(x9):
-1.44738811
```

-1.8
-1.6
-1.4
x10
-1.4
-1.6
-1.8
Solution by x=g(x) iteration
Touch: SPACE to iterate Table/STEP
1-9:dec.pl. 0:general format
Rescale New start Clean Quit

figure 9

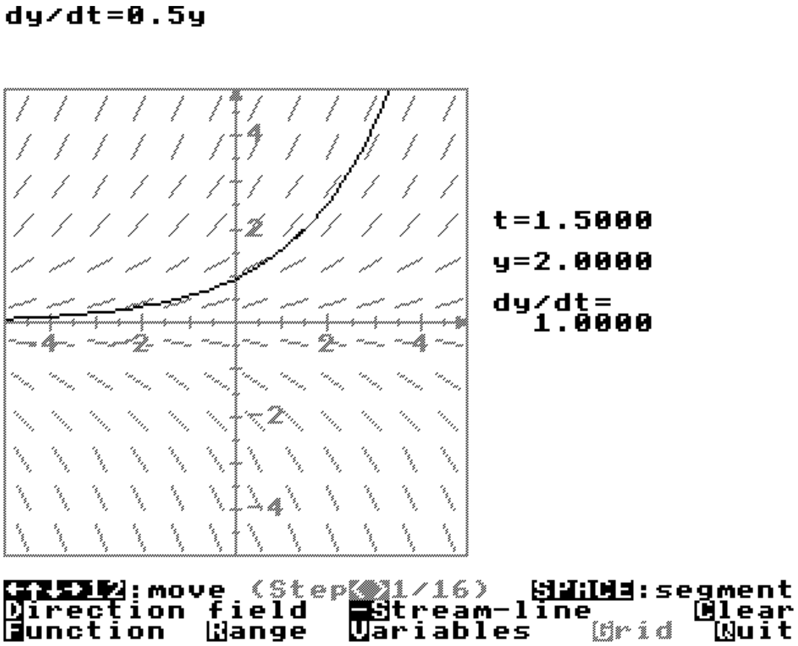
Further investigation of examples which converge and those which fail can lead to theories relating speed and success of convergence to the gradient of the curve y=g(x) at a root[11].

4.4 The Solution Sketcher†

The Solution Sketcher was designed for the School Mathematics 16–19 Curriculum to enable students to experience the process of building up a solution of a first order differential equation. A short line segment is drawn in the plane, with gradient determined by the differential equation. As the segment is moved, its gradient is adjusted. At any stage the current segment can be deposited onscreen and subsequent segments can be used to build up a solution. Thus the construction of a locally straight solution curve is enacted by the student. Other facilities include autodrawing of a solution and the drawing of a global array of segments to give an overall view of the total family of solutions (figure 10).

The naivety of the rudimentary construction process will soon be replaced by the greater power of step-by-step numerical methods, perhaps auto-drawn by the computer, and later by symbolic methods, as appropriate. But the cognitive insight of the building of a solution using little straight line-segments must not be underestimated. It lays foundations for subtle ideas about existence and uniqueness of solutions which can easily be visualized but are not well understood by pupils versed only in a symbolic traditional approach.

† Available only for the BBC computer, Archimedes, Nimbus PC. An IBM compatible program in the Graphic Approach to the Calculus draws direction fields and automatic solutions in a similar manner.



```
dy/dt=0.5y

t=1.5000
y=2.0000
dy/dt=
   1.0000
```

4
2
-4
-2
2
4
-2
-4
←↑→↓1/2:move (Step◆1/16) SPACE:segment
Direction field ↵Stream-line Clear
Function Range Variables Grid Quit

figure 10

4.4 The Parametric Function Analyser[†]

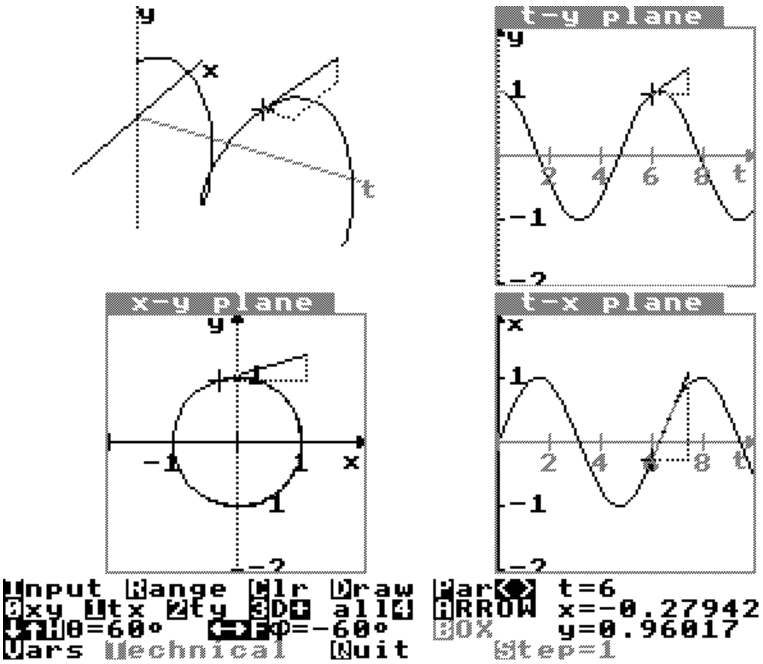
This program was designed for the SMP 16–19 curriculum to give new insight into parametric functions in the form x=x(t), y=y(t), not only in terms of the usual representation as a parametric curve in the x-y plane, but also the separate graphs x=x(t) and y=y(t) and the three-dimensional picture of the curve created by the points (t,x,y) as t varies. In addition, an arrow may be drawn through two close points on the curve with parameters t and t+h. For small h this approximates to the tangent at t (if there is one!) and the projections of the arrow are approximations to the tangent to the projected curve in the three planes t-x, t-y and x-y (figure 11).

The tangent vector has components (dt,dx,dy), so relationships such as

$$\frac{\mathrm{d}y}{\mathrm{d}t} = \frac{\mathrm{d}y}{\mathrm{d}x}\frac{\mathrm{d}x}{\mathrm{d}t}$$

are simply arithmetic relationships between the sides of a box in three space. The visualization even helps to give insight into the singular and difficult case of what happens when dx=0...

[†] Only available for the BBC computer, Archimedes and Nimbus PC.



t=6
x=-0.27942
y=0.96017
θ=60°
φ=-60°
Step=1

figure 11

4.5 Extensions to Second Order and Simultaneous Differential Equations[†]

When an attempt is made to extend the notion of building up a solution of a first order differential equation to a second order differential equation, then it seems to fail because there seems no longer to be a single direction to follow. Through any point in the plane there may be many solutions. Figure 12 shows three solutions of $d^2x/dt^2 = -x$ passing through the origin. There is a unique solution through this point for each starting direction.

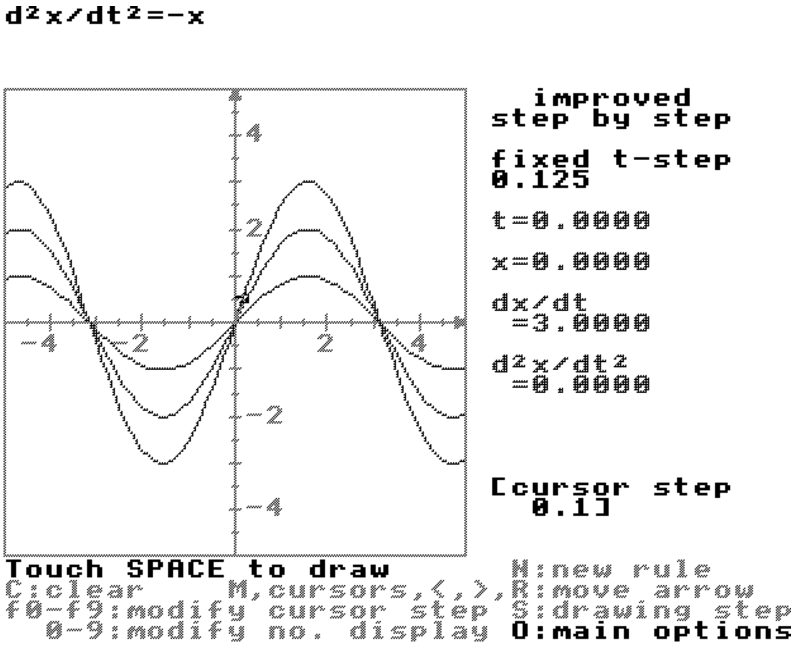
There is a subtle way of dealing with this which simply involves defining the new variable $v=dx/dt$ where v is the gradient of the t-x curve. The original equation becomes

$$dv/dt=-x,$$

and, taken with the definition of v, this gives the two simultaneous equations:

$$dx/dt=v$$
$$dv/dt=-x.$$

[†] Available for all IBM compatibles, the BBC computer, Archimedes, Nimbus PC.



```
d²x/dt²=-x

   improved
step by step
fixed t-step
0.125
t=0.0000
x=0.0000
dx/dt
 =3.0000
d²x/dt²
 =0.0000

[cursor step
  0.1]
```

figure 12

These give the direction of the tangent vector (dt,dx,dv) in three dimensional space as:

$$\mathrm{d}x = v\,\mathrm{d}t,$$
$$\mathrm{d}v = -x\,\mathrm{d}t,$$

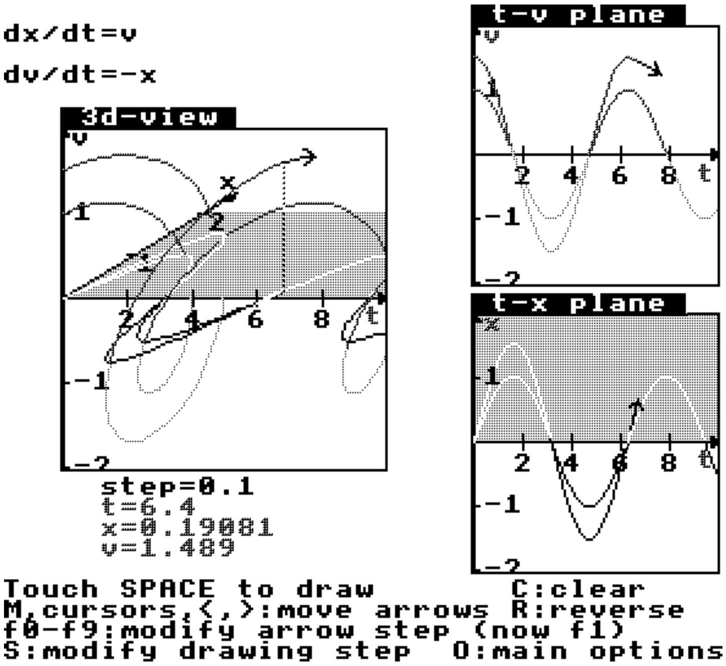
so

$$(\mathrm{d}t,\ \mathrm{d}x,\ \mathrm{d}v) = (\mathrm{d}t,\ v\,\mathrm{d}t,\ -x\,\mathrm{d}t),$$

is in the direction $(1,v,-x)$.

Again a unique tangential direction is given, but now in three-dimensional (t,x,v)-space. Through every point (t,x,v) in three-dimensional space there is a unique solution found by following the given tangential directions. In practice a good approximation can be built up by putting short line segments in the appropriate directions end to end (again a triumph for local straightness!).

The unique solution starting at $t=t_0$, $x=x_0$, $v=v_0$ projects onto the (t,x) plane to give the unique solution through the point $t=t_0$, $x=x_0$ starting with gradient dx/dt=v_0. In figure 13 the 3d-view has spiral solutions which do not intersect, but the projection onto the t-x plane reveals the intersecting solutions of a second order differential equation in figure 12.



```
dx/dt=v
dv/dt=-x
```

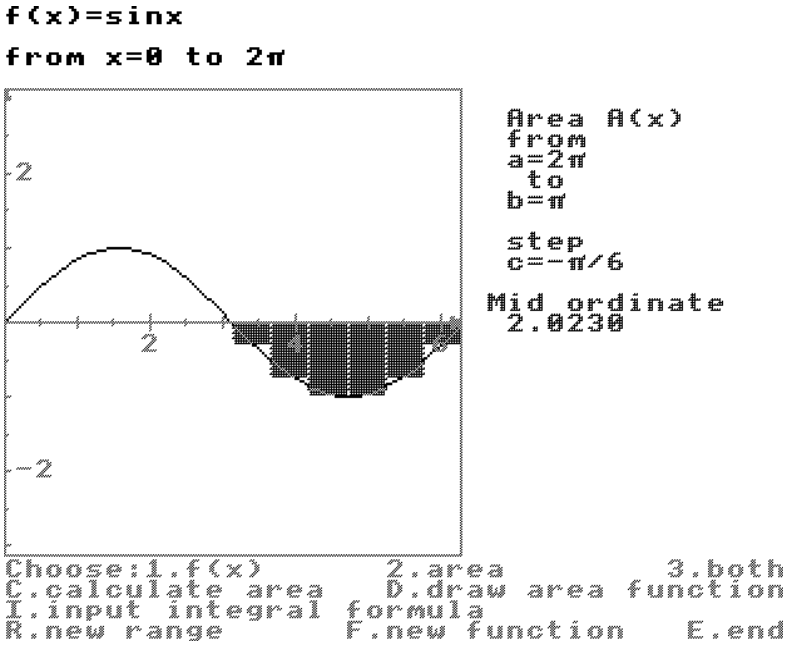
```
step=0.1
t=6.4
x=0.19081
v=1.489
Touch SPACE to draw       C:clear
M,cursors,<,>:move arrows R:reverse
f0-f9:modify arrow step (now f1)
S:modify drawing step  O:main options
```

figure 13

4.5 Area[†]

The area program gives the student the power to see the manner in which the area “under” a curve can be calculated using thin rectangular strips, or other useful rules, such as the trapezium or Simpson rule. Visually, by taking a larger number of strips, it becomes apparent that sum of strip-areas is likely to give the value of the area under the curve. However, as usual, the software is extremely flexible and may be used for all kinds of investigations [7,13]. For instance, few students understand the idea of the sign of the area calculation given in a traditional course, because it is left too late in the development and related to the symbolism of the integral, instead of being introduced from the beginning as a numerical calculation involving signed lengths. Most students, and not a few teachers, believe that the area is “positive above, negative below”, but if the sign of the step is negative, the reverse is true... (Figure 14 has a negative step and a negative ordinate in the range working backwards from 2π to π, giving a positive result when the curve is below the *x*-axis).

[†] Available for all IBM compatibles, the BBC computer, Archimedes, Nimbus PC.

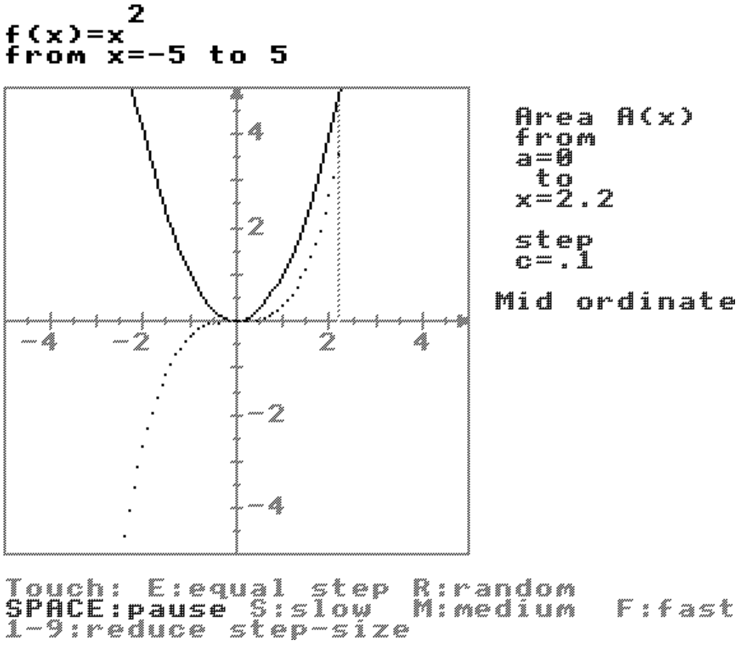


```
f(x)=sinx
from x=0 to 2π

Area A(x)
from
a=2π
 to
b=π

step
c=-π/6
Mid ordinate
 2.0230
```

2
-2
2
4
Choose:1.f(x)   2.area   3.both
C.calculate area   D.draw area function
I.input integral formula
R.new range   F.new function   E.end

figure 14

The area-so-far graph may be drawn by plotting the value of the cumulative area calculations from a fixed point to a variable point. Figure 15 shows the superimposition of two area calculations, first from x=0 to x=–5 with the negative step –0.1, then from x=0 to the right with positive step 0.1. Notice the cubic shape of the dots of the area curve, which experts will recognize as $y=x^3/3$. But we rarely consider this for *negative x*.



```
f(x)=x^2
from x=-5 to 5

Area A(x)
from
a=0
 to
x=2.2

step
c=.1
Mid ordinate
```

4
2
-2
-4
-4
-2
2
4
Touch: E:equal step R:random
SPACE:pause S:slow M:medium F:fast
1-9:reduce step-size

figure 15

4.6 The Function Analyser[†]

The Function Analyser is written for the SMP 16–19 curriculum. One of its many facilities is the ability to use *function notation* in the form f(x) and to combine and compose functions in the normal way. Thus it is possible to define and draw the graph of, say f(x)=sin$x$, and then to superimpose the graph of y=f(x)+2 (figure 16).

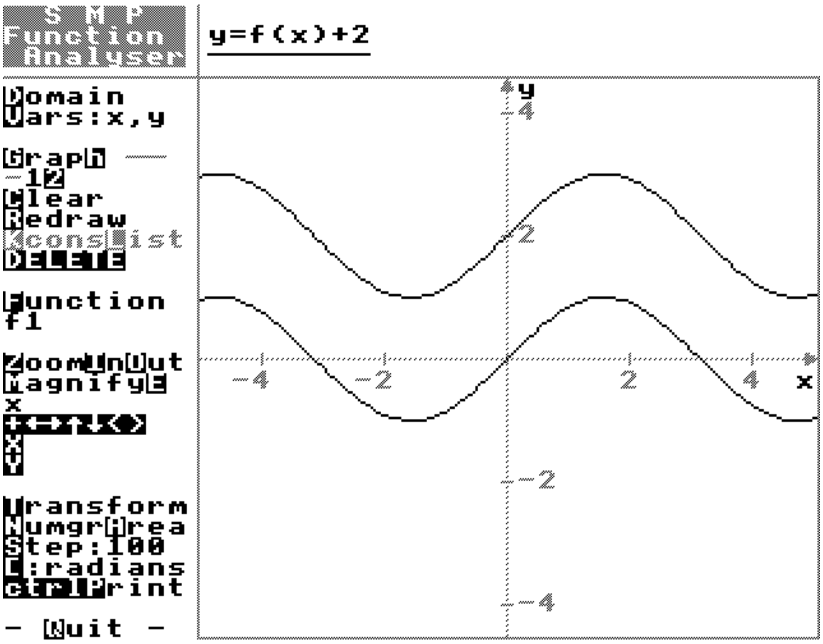


figure 16

One routine in the program allows a graph to be shifted or stretched horizontally or vertically by specified amounts. It is clear that the graph of f(x)+2 is the same as the graph of f(x) shifted up two units, but what shift is required to produce the graph of f(x+2), and why? ...

4.7 Numerical Gradients and Areas as Functions[‡]

Now that the 32 bit RISC chip is becoming freely available in British Schools at a reasonable price (where Britain leads, the rest of the world will follow or steal...), awesome power in calculation is widely available to children. There are many programs around that will draw numerical gradients or numerical areas, but these are usually just a sequence of points plotted on the screen. An area graph plotted as in figure 15 simply records the cumulative area calculations pictorially and does not remember them in any way. But the area under a graph of a given function, from x=a to x=b, with a given step, say h, is calculated by a straightforward computer procedure. Given an extremely fast processor this can be calculated almost instantaneously and may be considered as a function depending on the graph and the values of a,b, and h.

[†] Currently only available for the BBC computer, Archimedes and Nimbus PC.

[‡] requires the processor speed of the Archimedes 32 bit RISC chip to function fast enough, but also available at a slower speed for the Nimbus PC.

In the *Function Analyser* [12] the expression:

area(*expr*, *a*, *b*, *h*)

is interpreted as the area under the graph given by the expression *expr*, from $x=a$ to $x=b$ using step-width *h*. Thus:

area(sin*x*, 0,*x*,0.1)

is the area under $y=\sin x$ from 0 to *x* with step-width 0.1. Such is the power of the Archimedes that this graph can be drawn from $x=-5$ to 5 with 100 intermediate points in less than 3 seconds (using interpreted BBC BASIC), even though at the extremes each area calculation is adding up the areas of 50 strips for each point plotted.

Now that the numerical area is a *function*, it may be treated as any other function and numerically differentiated. Figure 17 shows the area function area(bl(*x*),0,*x*,*s*) under the blancmange function from 0 to *x* using strip-width $s=0.05$. (The area function is the rather bland looking increasing function, not the pudding-like blancmange).

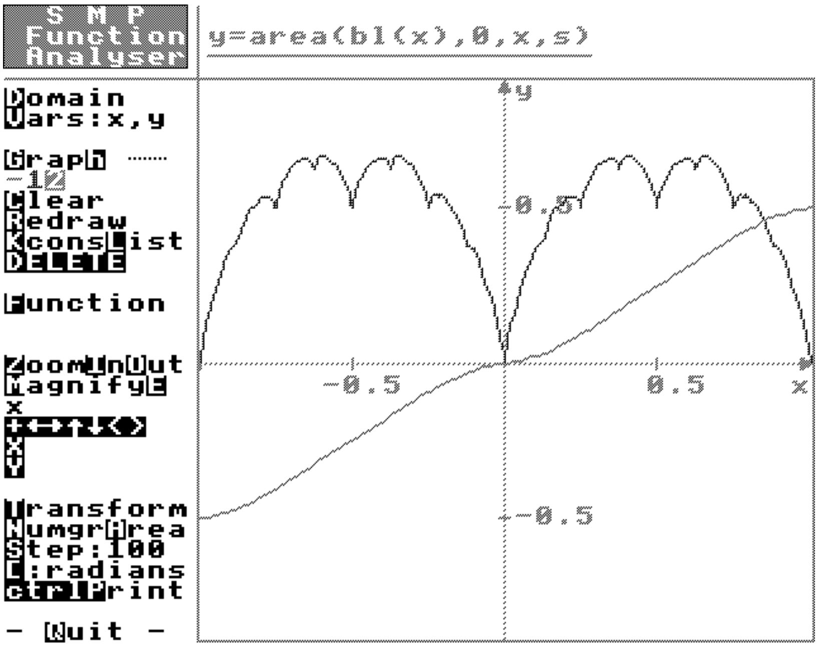


figure 17

Of course, this graph is not the exact area function, but it is a good-looking approximation to it – as good as one could hope to get on a computer screen. Notice that it looks relatively smooth – which it is, if one ignores the pixellation problem – because the derivative of the exact area is the blancmange function. The exact area function for the blancmange is a function which is differentiable everywhere once and nowhere twice!... At last, using the computer, the student has a chance of gaining insight as to how such things can happen... Mathematical analysis can begin to take on a real meaning...

The other graph in figure 17, by the way, may look like the blancmange function, but it is not! It was in fact drawn as the graph of

$$\text{area}(\text{bl}(x),x,x+w,h)/w$$

which is the area under the blancmange from x to $x+w$ (calculated using strips width h) divided by the width w, (here with $w=h=0.001$). Thus it is the numerical gradient of the numerical area of the blancmange function!

Even though the blancmange function is nowhere differentiable, its area function is quite smooth and differentiable everywhere.

One way of understanding what this means is to think in a versatile way about the meaning of area and the fundamental theorem.

Imagine a tiny part of the graph stretched horizontally (figure 18). In many cases the graph of a function stretches out to look flat – the more it is stretched, the flatter it gets.

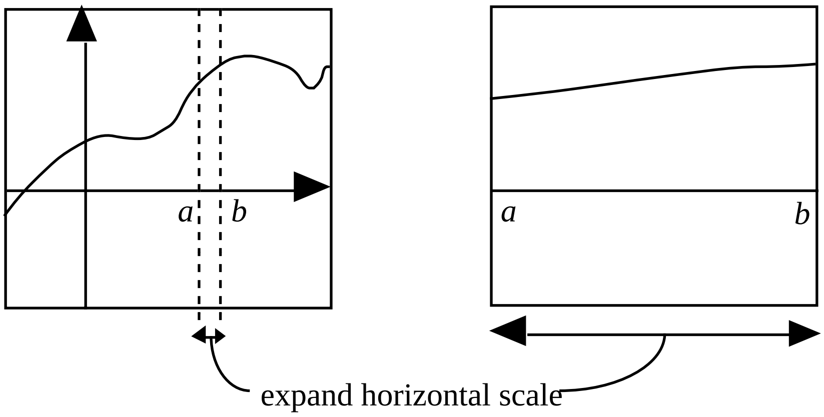


figure 18

This is easy to see with a graph program by changing the x and y scales – using a thin x-range and a normal y-range but drawing the graph in a standard graph window will stretch the thin x-range out horizontally.

Figure 19 shows that if the area from a fixed point a to a variable point x is $\mathrm{A}(x)$, then the area from a to $x+h$ is $\mathrm{A}(x+h)$, so the change in area from x to $x+h$ is $\mathrm{A}(x+h)–\mathrm{A}(x)$.

If the strip from x to $x+h$ is approximately a rectangle width h, height $\mathrm{f}(x)$, then its area is

$$\mathrm{A}(x+h)-\mathrm{A}(x) \approx \mathrm{f}(x)h$$

giving

$$\frac{\mathrm{A}(x+h)-\mathrm{A}(x)}{h} \approx \mathrm{f}(x).$$

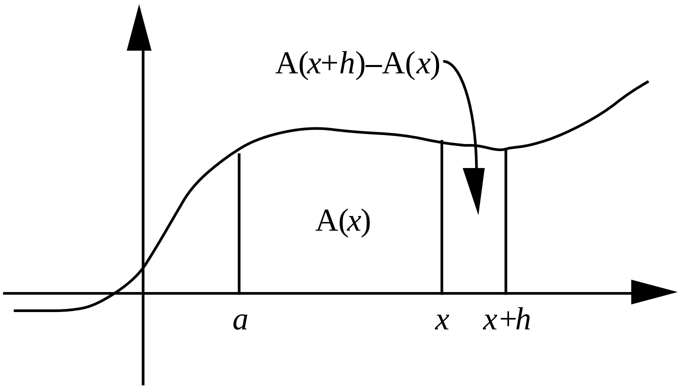


figure 19

As h gets smaller, the graph gets pulled flatter, the approximation gets better and so

$$A'(x) = f(x)$$

– the **fundamental theorem of the calculus.**

A formal proof of this requires the notion of continuity. This notion is usually confused intuitively by students, teachers and mathematicians alike. Ask anyone with some knowledge of the concept to explain what it means and the likely answer is that is is a function whose graph has “no gaps” – its graph can be drawn “without taking the pencil off the paper” etc, etc. These ideas are *not* the intuitive beginnings of continuity but of “connectedness” which is mathematically mathematically linked, but technically quite different.

An intuitive idea of continuity arises in the fundamental theorem of calculus. What is the essential property that makes the fundamental theorem work? It is that a tiny portion of the graph stretched horizontally looks flat. How can this essential property be characterized? The answer is that the computer picture does not draw a theoretical line, but a line of pixels of finite thickness.

Consider a simplified model of what is happening in stretching the graph horizontally. Suppose that graph picture has middle x-value $x=x_0$ and the point $(x_0,f(x_0))$ on the graph is in the middle of a pixel whose upper and lower values are $f(x_0)–e$ and $f(x_0)+e$. To fit the graph in a horizontal line of pixels means finding a small x-range from $x_0–d$ to x_0+d so that for any x in this range the value of $f(x)$ lies in the “pixel range” between $f(x_0)–e$ and $f(x_0)+e$ (figure 20).

The definition of continuity is:

> The function f is continuous at x_0 if, given *any* specified error $e>0$, there can be found a (small) distance d such that whenever x is between $x_0–d$ and x_0+d, so $f(x)$ is between $f(x_0)–e$ and $f(x_0)+e$.

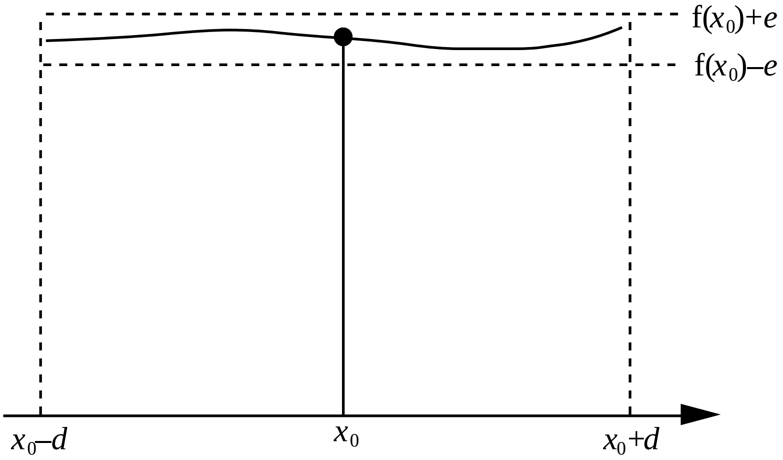


figure 20

Mathematicians usually replace the letters *e* and *d* by their Greek equivalents – ε (epsilon) and δ (delta). (Perhaps it makes them seem more imposing!). The definition is written formally as:

> The function f is continuous at x_0 if, given any ε>0, there exists δ > 0, such that $x_0–\delta < x < x_0+\delta$ implies $f(x_0)–\varepsilon < f(x) < f(x_0)+\varepsilon$.

The ramifications of this definition take months, even years, to understand in all its glory, but remember that it has an appealing cognitive root: *a continuous function is one whose graph has the property that any suitably tiny portion stretched horizontally will pull out flat.*

Thus we have again illustrated that the order of development may be very different from the traditional logical one. Here continuity has not been studied *before* the fundamental theorem – indeed it is very difficult for students to grasp in its logical form. Instead, continuity arises through exploration as a natural *requirement* for the fundamental theorem and is ripe for further investigation *after* the fundamental theorem has been encountered.

5. Reflections

In the preceding sections we have had the briefest acquaintance with a number of generic organizers involved with functions, graphs and the calculus. In every case the software is designed to expose one or more mathematical ideas with the dual role of being both immediately appealing to students and also providing foundational concepts on which the ideas can be built. By exploring examples which work and examples which fail, it is possible for the students to gain a far more versatile understanding. But they need help in identifying the subtle powerful ideas embedded in the software and formulating it in a way that can be used as foundations for the future. And here they need a teacher, as mentor, to guide and encourage them in their journey through the theory, a task made potentially much easier by the existence of the new software tools.

References

1. Blackett N. 1987: *Computer Graphics and Children's Understanding of Linear and Locally Linear Graphs*, M.Sc. Thesis, The University of Warwick.

2. Brumby M.N. 1982: ‘Consistent differences in Cognitive Styles Shown for Qualitative Biological Problem Solving’, *British Journal of Educational Psychology*, 52, 244–257.

3. Poincaré H. 1913: *The Foundations of Science* (translated Halstead G.B.), as reprinted 1982 by the University Press of America, Washington, U.S.A., page 35.

4. Tall D.O.1982: ‘The blancmange function, continuous everywhere but differentiable nowhere’, *Mathematical Gazette*, 66 11–22.

5. Tall D.O. 1985:*Supergraph* †, Glentop Press, London.

6. Tall D.O. 1986:*Graphic Calculus I : Differentiation* †, Glentop Press, London.

7. Tall D.O. 1986: *Graphic Calculus II : Integration* †, Glentop Press, London.

8. Tall D.O. 1986: *Graphic Calculus III : Differential Equations* †, Glentop Press, London.

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

10. Tall D.O. 1987:*Readings in Mathematical Education: Understanding the calculus*, Association of Teachers of Mathematics (collected articles from ‘Mathematics Teaching’, 1985–7)

11. Tall D.O. 1989:*Graphic Calculus IV: Numerical Solutions of equations* †, Rivendell Software, Warwick.

12. Tall D.O. 1989: *Real Functions & Graphs: SMP 16–19* †, Rivendell Software, reprinted by Cambridge University Press (1991).

13. Tall D.O., Blokland P. & Kok D. 1989: *A Graphic Approach to the Calculus*‡ , Sunburst Inc, USA.

14. Thomas M.O.J., 1988: A Conceptual Approach to the Early Learning of Algebra Using a Computer, Ph.D. Thesis, University of Warwick.

†Computer programs for the BBC, Nimbus PC and Archimedes Computers, plus book.

‡ 15 computer programs for IBM compatibles with 64 page booklet.