

Gerald's Column by Gerald Fitton

Well! I had expected some interesting letters as a result of my excursions into mathematical (and other) philosophical matters last month. I found them all most interesting. However, the range of subjects included in your correspondence to me far exceeds that which can be included in a computer magazine, even with the licence which Paul allows me. What I've done with the majority of this correspondence is to answer each of you individually – at least I think I have.

This month I shall correct a mistake from last month; I shall indulge myself with a digression or two including a brief introduction to a calculus you may not have heard of: and I start on the marathon task of making a comparison between the spreadsheet packages PipeDream, Eureka and Fireworkz. I hope that Paul will be able to arrange for me to receive a copy of Schema in the near future so that I can add that package to the range considered.

The Gamma function

Your remarks to me on this topic divide into two classes. The first class contains questions such as: “How did you produce those graphs using PipeDream?” Although I shall deal with that in detail another day, in this month's column I make some comparisons between line graphs drawn in PipeDream, Eureka and Fireworkz. The second class relates to three mistakes you reckon I made. By the way, I don't count Paul's mistake (not mine) of using a lower case *g* in “The gamma function” heading! (Nevertheless, I suppose I should take some of the blame since I invariably use too many capitals in my headings whereas I know that Paul's house style for headings excludes all but the essential ones.)

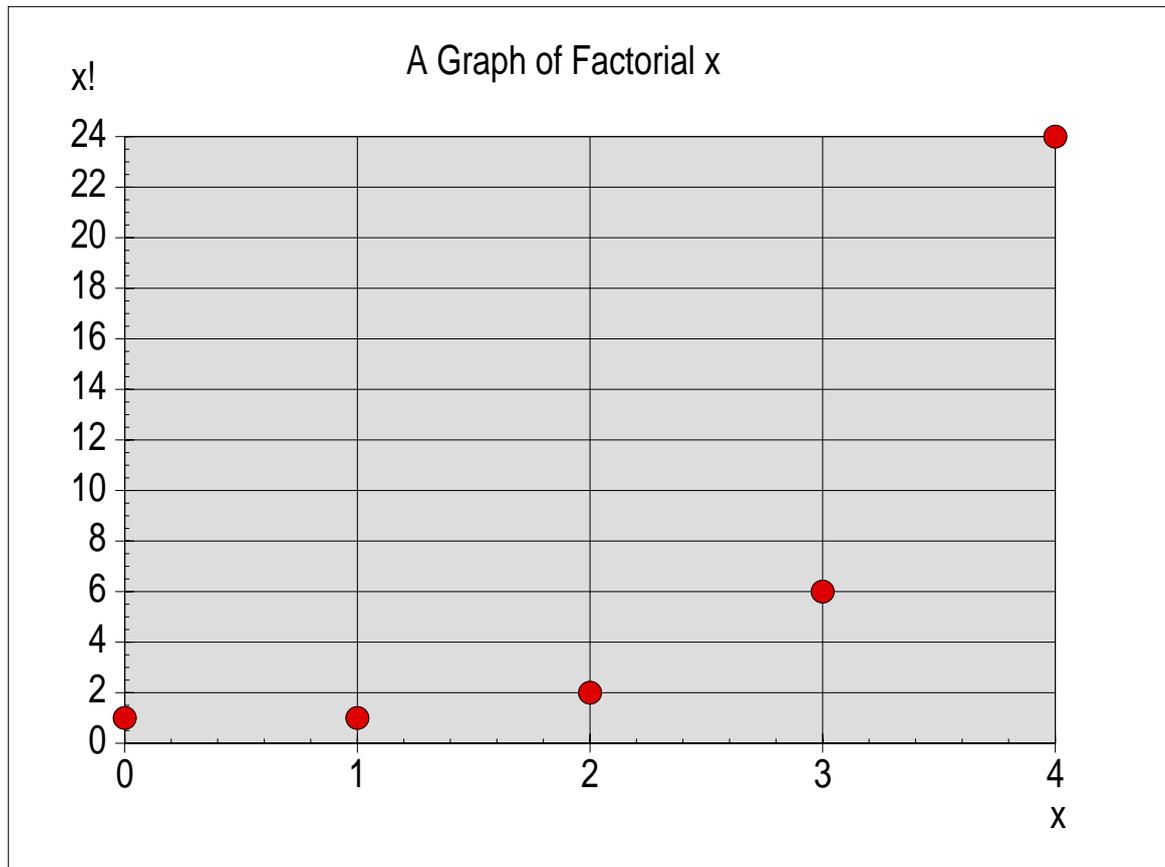
Last month I suggested that the Gamma function was symmetrical. I made a mistake! Even before the first letter came in I (somewhat belatedly) rediscovered my University notes (which I haven't looked at for a few decades) and, sure enough, I had got confused between the shape of the Gamma function and something else! That's what comes of relying too much on memories from long ago.

I would like your permission to count the two other ‘mistakes’ to which you drew my attention as over simplifications (which I knew about when I did them) rather than errors. Now I regret that I didn't take account of the mathematical sophistication of some of my readers. Let me correct those two mistakes as well!

Let me remind you that I defined the factorial function with the example of factorial 7: $7! = 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1$. I included the graph of the function $y = x!$ shown below. You will see that it consists of a set of points with nothing in between (because the factorial function exists only for positive integers). The ‘mistake’ was that I implied that the values given by the factorial function at integer values of x were the same values that you would get by evaluating the Gamma function. The truth is that $\text{Gamma}(x) = (x - 1)!$ so that the graph of the Gamma function is shifted over by one unit to the right.

My other ‘mistake’ was to say “Loosely translated a regular function through a few given points is the function giving the smoothest possible curve drawn through those points.” To be strictly accurate my words “a few . . . points” should have read “an infinite number of

points"! My reason for not referring to an infinite number of points is that I thought too many people would have translated "an infinite number of points" as "all the points there are". Quite rightly I have been taken to task for being slipshod.



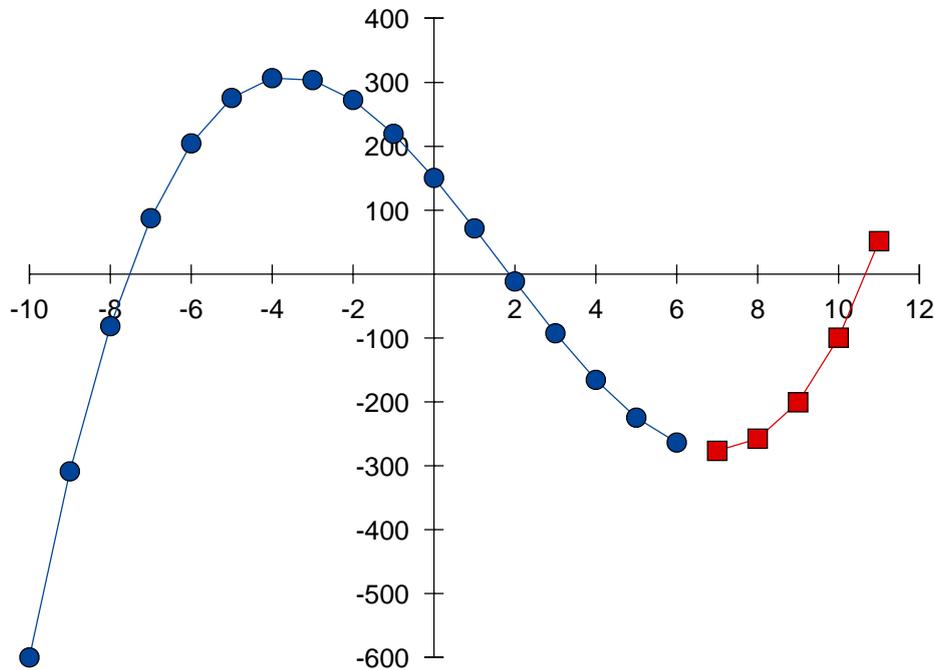
Please allow me to expand. There are too many misconceptions about the number of items which go to make up an infinite set. The number of integers is infinite but the integers don't account for all the numbers there are. If you know the value of a function (such as the factorial function) at an infinite number of values of x (in our case 'just' the integers) then there are many other points on the x - axis (the so called number line), such as the fractions between the integers (eg at $x = 0.5$), which are still available and at which the factorial function doesn't exist. You can 'fill in' the blanks in your knowledge of the function in a *unique* way using the mathematical concept called 'regularity' (which, being a bit lax about it - but descriptive - I've called smoothness). The Gamma function (after shifting along the x - axis) is the natural extension of the factorial function to every single point that there is on the number line (and beyond the number line to complex numbers including the square root of -1). Some day (but not today) you might persuade me to write a few words about infinities of different 'sizes' - those infinite sets which can be counted with the integers and those that can't because there's too many members in the set to be counted using just the integers (so you have to use the number of points on a line instead).

The calculus of finite differences

It is possible to calculate the values of the Gamma function for positive real numbers by interpolation between the factorial values using a spreadsheet to do the sums. The subject you need to look up in your numerical methods text book is "The calculus of finite

differences". As an example of the use of this calculus I will show you how to extrapolate values of a function which you suspect might be a cubic. All files referred to in this section appear on the monthly disc.

The drawfile shown below has been generated using Eureka.



For the purpose of this exercise I want you to assume that the points appearing on this graph as circular dots (the range from $x = -10$ to $+6$) have been obtained by experiment. Assume the values are approximate and that you suspect the function is a cubic. Your task is to extrapolate from the known approximate values to larger values of x . I have shown the points you have to find as square markers on the graph.

The screenshot below shows the Eureka sheet which I have used to calculate the finite differences and could have been used to make the extrapolation.

Let me admit that I've cheated and calculated all the values in the B column using the formula $y = ax^3 + bx^2 + cx + d$ with the values of the parameters (a, b, c and d) taken from B5B8. For the purpose of my explanation you have to assume the values of y corresponding to $x = -10$ to $+6$ are available (experimental results) but not those beyond $x = 6$. The so called 'finite differences' appear in columns C to F in the columns marked Del_1 onwards. The convention for finite differences is that the column heading I've called Del_1 should have been written as Δx but I've been too lazy to introduce the MathGreek font which is needed into the Eureka sheet. 'Del' is short for the Greek capital 'Delta'. The formula in C11 (in Eureka format) is: $=B12 - B11$. Check the arithmetic and you'll see that it evaluates as $-309 - (-600) = +291$. I copied this simple formula down and to the right through all the finite difference cells in which you see values.

You will notice that by the time we get to column F all these differences have become zero. It is the characteristic mark of a cubic that the fourth set of differences are zero. Similarly if your data satisfies a fifth power equation then it would be the sixth set of differences

which would drop to zero.

	A	B	C	D	E	F
1	Extrapolation – using the Calculus of Finite Differences					
2						
3	The cubic $y = ax^3 + bx^2 + cx + d$					
4						
5	a =	1				
6	b =	-6				
7	c =	-75				
8	d =	150				
9						
10	x	y	Del_1	Del_2	Del_3	Del_4
11	-10	-600	201	-64	0	0
12	-9	-309	227	-58	0	0
13	-8	-82	169	-52	0	0
14	-7	87	117	-46	0	0
15	-6	204	71	-40	0	0
16	-5	275	31	-34	0	0
17	-4	306	-3	-28	0	0
18	-3	303	-31	-22	0	0
19	-2	272	-63	-16	0	0
20	-1	219	-89	-10	0	0
21	0	150	-79	-4	0	0
22	1	71	-83	2	0	0
23	2	-12	-81	8	0	0
24	3	-93	-73	14	0	0
25	4	-166	-59	20	0	0
26	5	-225	-39	26	0	
27	6	-264	-13	32		
28	7	-277	19			
29	8	-258				
30	9	-201				
31	10	-100				
32	11	51				

I'm sure that you'll agree that the column containing the zeros is very easily extrapolated to larger values of x . Just fill F26 to F32 with zeros. Do this and then work steadily from right to left filling in the blank slots with a formula such as $C29 = C28 + D28$ (which can be copied into all the remaining blank slots), until you come to column B (pretend the slots B29 to B32 contain blanks which you fill with copies of the same formula). The set of values which you get in column B is the extrapolation of the known data and will produce the points depicted as squares in the drawfile of the Eureka chart shown above.

Of course, you may not know whether the approximate experimental data is a cubic or whether it is something else. The practical way of dealing with real data is to continue to create columns of finite differences until you have a column in which the numbers are

fairly close to zero (but with some functions it doesn't happen – that makes it a bit tricky). It does take some practice to choose the correct column and the tendency amongst those without experience is to go too far to the right (too many columns). If you go too far to the right then the column of differences will increase instead of getting smaller (as they should). Extrapolating a set of experimental data using this (add zeros to the last column) method relies on the function being a regular (smooth) function and not one with a 'kink' in it somewhere outside the range of values you know.

If you are interested in learning the art of choosing the column to fill with zeros then I suggest that you experiment with a column of values which contains an analytical function such as a cubic plus a small \pm random error.

I have not explained how to use finite differences for interpolation (finding values between the known data points) – (a) let's leave that for another day and (b) see if you can do it first! What I would draw to your attention is that finite differences are a powerful way of smoothing out the experimental errors in a set of data points. The well known linear regression process for drawing the 'best straight line' through a set of points is easy but drawing a best curve is much harder! You can also find a numerical value for the slope of the curve (the first derivative) using finite differences (and this is the clue you need to find the interpolation formula).

It can be shown (using mathematical logic) that any regular function – such as the power function or even the $y = \sin(x)$ function – can be approximated by a polynomial. Indeed, if you take an infinite number of terms, then your polynomial will produce exactly the same values as the regular function. In terms of the calculus of finite differences this means that if you use enough columns then you can represent any regular function in terms of its finite differences and hence extrapolate it – or interpolate between known values, to any required degree of accuracy.

The calculus of finite differences has a long history (possibly more than three hundred years) as a numerical method for solving the problem of extrapolation from and interpolation between known measured data values. In my view spreadsheets make the use of finite difference techniques so easy that they are something which could be taught at school rather than as at present, at university level only.

Finally, let me challenge you to an interpolation problem. Assume that you have been given as many of the integer powers of 2 as you need (in column B of a spreadsheet). Use the calculus of finite differences to interpolate this 'integer powers of 2' function, $pwr(2,n)$, to find the value of $pwr(2, 0.5)$, the irrational square root of 2, to about four decimal places. I must point out to you that this is one of those tricky problems where the finite differences don't get smaller as you move to the right in the spreadsheet – so it's not an easy one.

The smooth answer to puzzle number 50

In the answer to Colin Singleton's puzzle number 50 he shows that the identity:

$$f(x) = \text{sqr}(1+f(x+1)) = (x+1)$$

is true for all positive integers (the so called 'Natural numbers'). Many of my correspondents have thought hard about my suggestion of smoothing out such discrete

functions and then suggested the following:

Plot a graph of the function, $f(x) = \text{sqr}(1+f(x+1)) = (x+1)$, for a few natural numbers. Look at it and I'm sure that you'll have to agree (with my correspondents) that when you apply the concept of regularity (smoothness) to extend Colin's infinite set of discrete values to any number on the number line, it is inconceivable that the smooth curve is anything but the line $y = (x+1)$.

As I said last month, for those of you into complex numbers, the concept of regularity can be used to extend functions into the Complex number domain. Although I got several answers to the Real number part of the puzzle I got no takers for its extension to Complex numbers.

The scientific method and things eternal

I hope that Paul will allow me the space to answer publicly the question on this topic which appeared most often in this month's correspondence. It was a question which I anticipated because I've been asked it so many times before. You ask, "How is it possible for someone (such as me) who believes almost fanatically in logic (mathematics) and the scientific method (and is 'obviously intelligent') to rubbish them in the next paragraph?"

My short but enigmatic answer is: "It seems logical to me!"

My longer answer is that it can be proved (using logic) that the answers to some deep mathematical (and scientific) questions can be answered only by reference to a super-set of, let's call them values, truths or beliefs, taken from outside mathematics (and science) but within which the principles of mathematics and science must operate. By this I mean that in order to prove that mathematical 'truths' are really true (whatever that means) it is provable that you need a super-set of truths which go beyond mathematics (some mathematical philosophers have suggested logic is such a super-set). In turn it can be proved that this super-set needs a super-super-set and we're into an infinite sequence. What we need is an exit condition. During my life my belief in what forms a suitable exit condition has evolved from the simplest "I think – therefore I am" to those I hold today.

I continue with my longer answer by repeating a line from my last month's column (not my words but two quotes I've cribbed from elsewhere): "Now I see through a glass darkly" (meaning that currently I have only a limited grasp of Eternal values – which I believe contain the exit condition from the logical paradox of the previous paragraph) and "I (shall be able to) put away childish things" (meaning the concepts I can handle now such as causality and even logic) when I understand the nature of Eternity and things Eternal much better than I do at present. So, just as it took me a long time to understand the mathematical concept of Infinity, so one day (or maybe it will be when I have no more days) I hope to understand more of those things which are Eternal and thus discover an exit condition from the logical paradox which starts with the logically provable fact that mathematical 'truths' need non mathematical super-truths to validate them.

My comment to you (intended to be helpful as well as illuminating) is that I don't experience the paradox you do because I accept that, by its very nature, the scientific method is self limiting. I know one can't argue from analogy (unless it is provably a mathematical isomorphism) but analogies sometimes help the process of understanding.

My analogy is with the recursive definition of a function which has its own built in limitation – you must know when to stop defining the function recursively or you’ll finish up with nothing. In my search for the truth (in the world, the universe and everything) I have recognised a point (at least approximately) at which logic and the scientific method have reached their in-built limit of usefulness and I accept that I have to try something else outside the realm of logic if I am to proceed and grow in knowledge and wisdom.

Eureka – Schema – PipeDream – Fireworkz

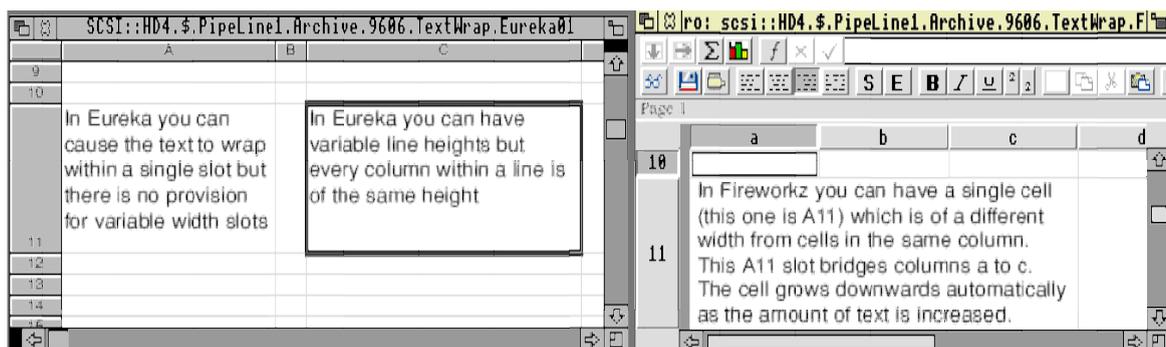
Thanks to Paul, Longman Logotron have sent me a copy of both Advantage and Eureka. Thanks also to those of you who’ve got Eureka and written to me about it – and to those of you who have Schema and want me to include it in my comparison. I have asked Paul if he can arrange for me to have a copy of Schema for that purpose.

Looking at the overall subject of comparing spreadsheets, it is not going to be possible for me to say everything I want to say in just one article so I’ve decided that I’ll deal with the different sorts of things you might want to do with a spreadsheet over a period of time with some sort of summary later on. Each of you will have different requirements so that something ‘missing’ from one package may be of no importance to you and may well be compensated for by other features which that package has but the others haven’t.

Line wrap

PipeDream and Fireworkz are primarily spreadsheets but they include word processing features. Eureka is a dedicated spreadsheet (without word processing features). The most important single feature of PipeDream and Fireworkz which makes them usable as word processors as well as spreadsheets is the line wrap feature. By this I mean that when you get to the end of a line in PipeDream or Fireworkz the text wraps around onto a new line (just like a word processor). Every slot or cell below moves down automatically and, where necessary, formulae are changed to reflect the existence of the additional rows.

The PC spreadsheets Lotus and Excel don’t have this line wrap facility. You can type across slots but, if you need a new line, you have to create the space by inserting rows manually and you have to deliberately drop down to the slots in the next row. This causes a lot of painful rewrites when you want to add a few words to or delete them from an earlier row. Eureka is better than Excel because you can invoke an option to wrap within a single slot but, unlike Fireworkz, you can not vary the slot width down the page. I hope the screen shot below clarifies what I mean.



In both Fireworkz and Eureka if you vary the width of the slot containing the text then the text will reformat automatically within the available space. In Fireworkz the slot height also changes automatically so that as you make the slot narrower it gets deeper; this ensures that all the text in the slot continues to be displayed.

PipeDream handles line wrap differently again. Every slot has two margins. The first margin is the slot boundary; the second, called the wrap margin, is the point at which that row wraps onto the next line. Paragraphs in PipeDream can run across cell boundaries up to the line wrap margin. The screen shot below demonstrates this feature. Line wrap occurs at the vertical down arrow visible at the right of column C. The position of this wrap margin can be adjusted for each column independently.

	A	B	C
3	x	y = x ³	y = x ^(1/3)
4	0.0000	0.0000	0.0000
5	0.1000	0.0010	0.4642
6	0.2000	0.0080	0.5848
7	0.3000	0.0270	0.6694
8			
9	These four rows of this PipeDream		
10	spreadsheet act as a single		
11	paragraph of text reformatting		
12	automatically as text is added or		
13	deleted to earlier rows.		

Now, if you are quite happy to do your word processing in, say, Impression, then the 'limitations' of Eureka's line wrap facility won't matter to you at all. Indeed, Eureka supports an OLE (Object Linking and Embedding) facility to Impression so, if you use the combination of Eureka and Impression then it's almost as if you have a live Eureka spreadsheet within Impression – something which many of you see as a superior way of doing things because you like Impression's word processing (or DTP) features more than those of Fireworkz. Neither PipeDream nor Fireworkz support OLE to Impression. I have Publisher V 4.09 – Publisher doesn't need a dongle any more – and Eureka's OLE to it works a treat!

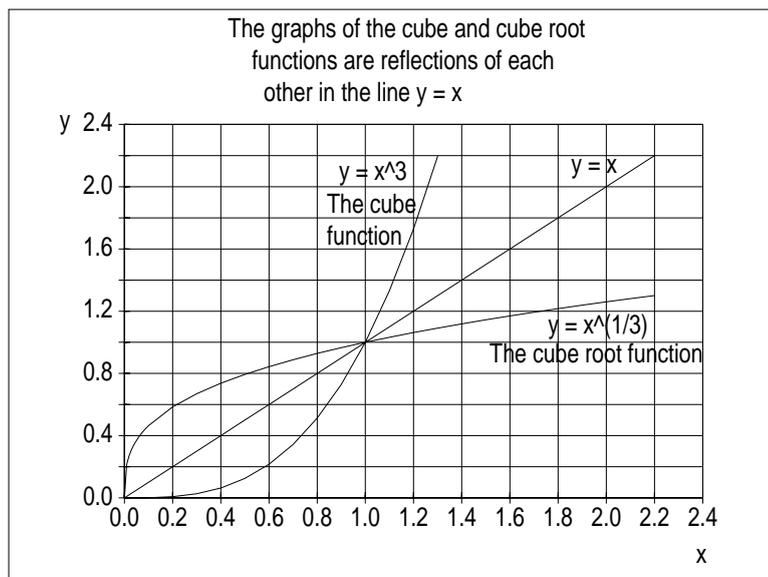
Graphs in PipeDream Eureka and Fireworkz

I've been asked to explain why, last month, I chose to produce the cube/cube root graph in PipeDream rather than in Fireworkz. Let me use that request as a starting point for my second comparison between the three spreadsheets listed above. You will find copies of all

the files and live charts of this section on the Archive monthly disc.

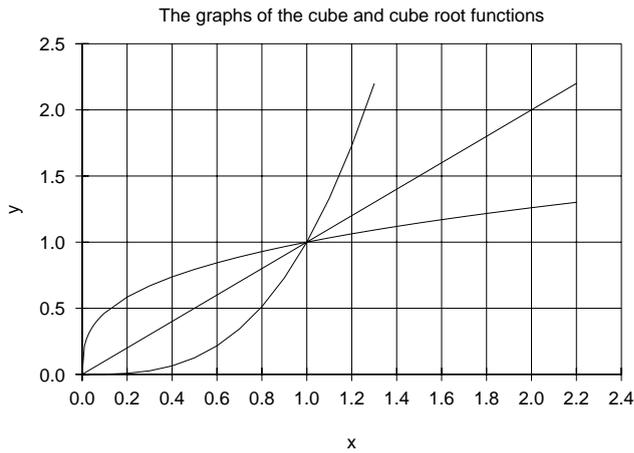
First of all I must say that, in my view, of all the spreadsheets I've tried out at home and at work (on a PC), PipeDream has the best charting features. Although the graph shown below is a drawfile, when running in PipeDream the chart is live. By this I mean that, if changes are made to the data, then the graph changes to match the new data.

You might not want all the charting features which PipeDream has. Let me list the ones demonstrated in the graphic shown below. There are three independent lines on the chart. You can add more lines without starting again. Each line has different x and y ranges. There is no limit to the number of text objects which you can add to the chart. These text objects can be placed anywhere you like.

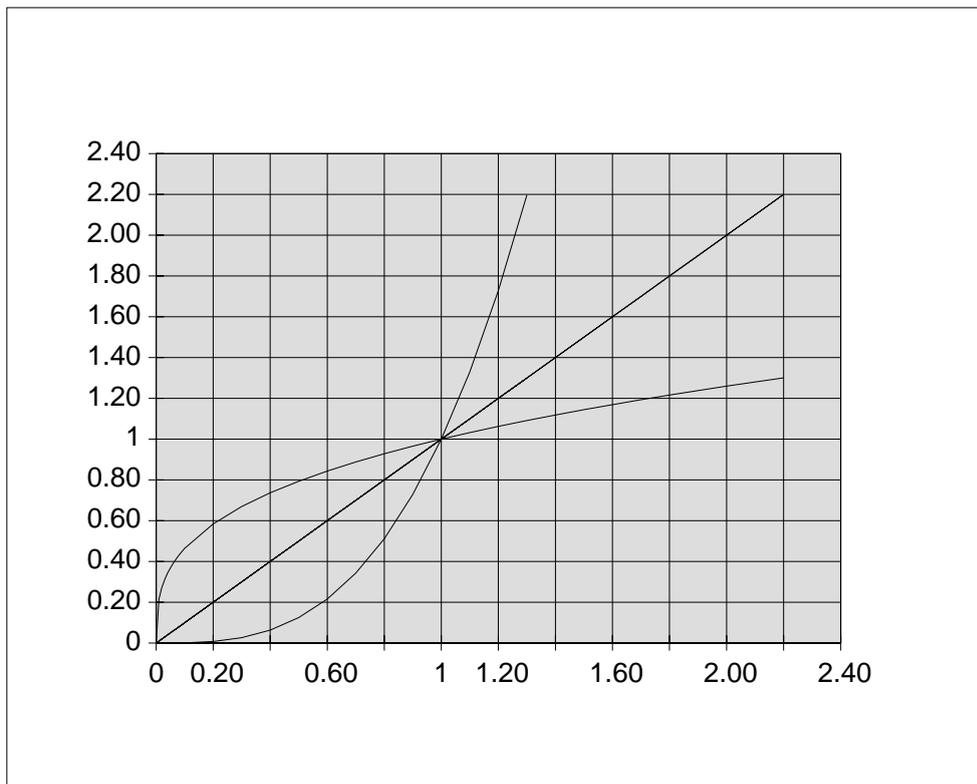


Below I include a screenshot showing how far I've got towards producing a similar chart using Eureka. I couldn't put the x and y where I wanted to; I would like the y to be turned through a right angle and placed at the top (see the PipeDream drawfile); I couldn't find a way of entering all the text I wanted to – particularly I wanted a multi line title.

Perhaps some of you with more experience than I have of Eureka will demonstrate to me (with a disc file please) that you can have a live Eureka chart with more of the features demonstrated in my PipeDream chart than I've been able to produce.



I have version 1.25/01 of Fireworkz and the screenshot below shows how far I could get:



Although the inconsistency in the number of decimal places on the axis annoys me (look at “1” on this chart compared with “1.00” on the others) and in spite of the difficulties of trying to get the numbers on the x – axis to appear at regular 0.2 intervals (which has driven me to distraction – you’ll see that some appear at 0.4 intervals) it is my inability to add any text without overlaying drawfiles over the chart which has made me abandon the use of Fireworkz for live charts. Of course, if you are willing to save the chart as a drawfile and then do some more work on it, OK! I usually want my charts to stay alive so that I can update them every month when new data comes along.

If you do not need all the features of PipeDream's charting package I have demonstrated then maybe you would prefer some of the features present in Fireworkz or Eureka (or Schema – I'll let you know about Schema if I get a copy) but which are lacking in PipeDream. For example, Fireworkz is much more WYSIWYG, it's got Styles; Eureka will produce excellent horizontal bar charts (a feature missing from both PipeDream and Fireworkz) and has some built in functions that Fireworkz hasn't got. Conversions between Eureka and the usual PC spreadsheets (Lotus and Excel) work much more smoothly than from Fireworkz.

My advice to you is this. If you're sure you don't and won't need the unique charting features of PipeDream then you should consider the advantages Fireworkz and Eureka have by way of Styles, portability or ease of use.

Finally

Thank you for all your letters. Please write to me (with a disc showing examples if appropriate) at the Abacus training address given at the back of Archive. I shall be particularly interested to hear about your experiences with either Eureka or Schema.