Doing Mathematics With New Tools: New Patterns of Thinking

David Tynan Kaye Stacey Gary Asp John Dowsey

University of Melbourne

This paper discusses new directions for mathematics in school, emerging with use of technologically advanced tools. The data are observations made during work with function graphers in the Technology Enriched Algebra project. There are three propositions. Students will use a wider range of notations, mixing them more than at present. Differences in tools will promote the development of different concepts. The variety of approaches to problems will increase markedly, due to apparently small differences in the design of software tools.

The Technology Enriched Algebra [TEA] project has been conducted by Gary Asp, John Dowsey, Kaye Stacey and David Tynan at the University of Melbourne since 1991 with support from the Australian Research Council. It has investigated the ways in which spreadsheets and function graphers can be used to enhance students' understanding of secondary school algebra. Results of the project have been reported at previous MERGA conferences (Asp, Dowsey, Stacey, 1992,1993) and a resource book is now in press. This paper arises from observations that we have made whilst preparing curriculum materials for students and trialing them with a variety of function graphers, and also from other research within the University (Steele, 1995).

In this paper, we draw together our observations about how the tools that are available to do mathematics affect patterns of mathematical thinking. We put forward the hypothesis that, as use of technology expands in schools, the number and variety of ways in which students approach a problem will increase and that this increase will be due in part to apparently small differences in the design of the software tools. As we show below, minor variations in the capabilities of function graphers lead their users to develop significantly different approaches to solving problems. We also propose that the variety of notations which students will use for mathematics will increase significantly. The examples in the paper are drawn from use of function graphers. However we expect that the use of other technological tools (e.g. statistical software, symbolic algebra) will raise similar issues.

Several empirical studies have been conducted as part of the TEA project and the results have contributed to the thinking behind this paper. However, this paper is primarily a reflective account of some of the differences which we expect to see in the practice of school mathematics when technological tools are more widely used. Our data at this stage is exploratory and is from two sources, the first being observations of students interacting with software whilst trialing our materialsand the second our own experiences. In the final stages of preparation of a resource book for function graphing, three of the investigators worked through the resource book, doing every part of every question as a conscientious student would. We each used a different function grapher: ANUGraph (Macintosh), Graphmatica (Windows) and the TI-82 graphics calculator. When we had finished we compared solutions in detail and found that we had developed quite different routine approaches to the problems. We hypothesise that in large measure the differences, which are explained in the paper, had developed in response to apparently small differences in the facilities offered by the function graphers.

The Need For A Highly Flexible Approach To Notation

Mathematics has a complex set of notations in common use. The school curriculum has in the past carefully controlled the notations which students use at each stage of schooling. With functions, for example, students first learn to use the x-y notation (e.g. $y = x^2 + 3x - 1$) where y and x have special roles. Later they learn to use 'function notation' (e.g. $f(x) = x^2 + 3x - 1$). Gradually different letters are used in place of the canonical x,y and f, often reflecting tradition in a particular situation (e.g. t for time, s for displacement); particular letters come to have special connotations (e.g. n for an integer and a,b and c for arbitrary constants); and refinements such as the use of suffices are introduced.

We began the TEA project believing that students may have to learn to deal with a new system of notation (e.g. function notation in Year 8) but not anticipating the notational deluge that use of function graphers actually causes. The three function graphers that we have trialed use different notations for functions. ANUGraph requires students to use function notation, calling functions f(x), g(x), $f_1(x)$ etc. The name of the function (with or without a suffix) can be specified but the variable must be x. (Parametric equations and relations are not considered here). The TI-82 uses x-y notation with suffices, so that different functions must be called y_1, y_2, y_3 etc and written in terms of x. On the other hand, Graphmatica also uses x-y notation but does not distinguish between different functions - they are all called y. All these function graphers permit the use of literal constants for writing families of functions and they specify the letters that can be used. The function graphers have their own codes for specifying information about functions. For example the Graphmatica line

y = b * x^a * e^(-c*x) {a: 0.2, 2.6, 0.4} {b: 5} {c: 3} {0, 100}   [1]

plots the family of functions specified for values of a from 0.2 to 2.6 moving in steps of 0.4, with b = 5, c = 3 and for values of x between 0 and 100. The letter a is the only letter that can be used in this way. The letters b and c can also be used as parameters, but they can take only one value per line.

Working with just a pencil and paper, students generally use whatever notation the question specifies to find the answers. (There are some obvious exceptions such as graphing functions called f and g both on an axis called the y-axis). Students do not expect to have to change the notation themselves. We now believe that this will not be possible in a technologically rich classroom. Students will have to adapt, changing the names of variables and constants and interchanging function and x-y notation as required by their technological tools. In the new environment, access will be through a notation specified in advance by a software supplier. An example of this is provided by the 1995 Victorian Mathematics Method Project, which asks students to investigate functions of the concentration (x mg/l) of a drug in the blood after t seconds. The project gives the general form of the function as

$$x = A\, t^b e^{(-ct)} \text{ for } t > 0.$$

To use Graphmatica to study the effect of changes in b on the given function, students must change the equation into the form shown in line [1]. The letter x must be replaced by y, t by x, A by b, b by a and c must stay the same. Other function graphers require different transformations. We began developing TEA materials expecting to give students problems written in the notation that they would use to solve the problem. We have now abandoned all hope of this. Most students in the near future will probably have to deal with at least two function graphers (e.g. a personal graphics calculator and a computer program which provides good printed output). To do this, they will have to be much more flexible.

Our trialing has led us to believe that students may adapt quite well to these demands. However, we do not have good data on this and it is an important question to examine in future trials. The answer to this question will have important practical implications: must a school specify the function grapher that students purchase or can students choose on the open market, should a school system commission special software for schools, will textbook writers need to tie their series to particular software?

How Differences In Tools Promote Differences in Concepts

We noted above that different function graphers use different notation for functions. We believe that it is likely that these notations and the ways in which they can be operated upon may promote different conceptions of what a function represents.

In the research literature, much has been written about the way in which functions are viewed as processes and as objects. Functions (and according to Piagetian theory, all other mathematical objects) begin as actions, develop into processes and then mature into objects (see, for example, Dubinsky, 1994). One of the tasks of mathematical experiences at school is to encourage the encapsulation of processes into objects. The function with rule $f(x) = x^2$, for example, begins as a shorthand for the process of squaring numbers, but eventually is constructed as an object in its own right with properties of its own.

Function notation, such as is used by ANUGraph, seems likely to encourage the development of the view of a function as an object. However, ANUGraph does not permit the operations on functions that are characteristic of them being seen as objects: for example pointwise addition and function composition. In contrast and despite its use of x-y notation, the TI-82 does allow the user to carry out these operations, so that a user can enter expressions such as $y_3 = y_1 * y_2$ for pointwise product or $y_3 = y_1(y_2)$ for composition.

There is currently a great deal of work designing multi-representational software that facilitates a richer development of a student's conception of function. We expect that the notation used by the software, and the operations on functions that it supports easily, will contribute to the nature of the concepts that students develop.

How Tools Affect Problem Solving Approaches

It is obvious that the ways in which we approach problems depend very much on the tools (physical, technological and intellectual) that we have available. For example, some approaches are not available to a person simply because he or she does not understand the standard mathematical procedures that others have. We appreciate that there are generally many ways in which a mathematical problem can be solved, but in our work with technology tools, we have observed that the number of options increases markedly. Below, we illustrate just one of the ways in which this happened as we worked with our three function graphers.

The problems used in our resource book were generally set in realistic contexts and used realistic numbers. Many of the problems required students to find an appropriate viewing window and obtain answers (e.g. points of intersection) from this. As it happens, realistic problems are not generally solved using viewing windows centred around the origin with a scale like the default scale, and in many problems, the main solution effort goes into locating an appropriate viewing window.

We found that each of us independently developed a particular approach to locating the appropriate viewing rectangle. These different approaches, which are illustrated below, seemed to develop in response to the different options that are provided for working with the viewing window.

OZ-Mobile Phones

OZ-Mobile is a new mobile phone company, offering customers three charging schemes:

OZ-Mobile Scheme A $10 per month and a call charge of $1.20 per minute

OZ-Mobile Scheme B $20 per month and a call charge of $0.80 per minute

OZ-Mobile Scheme C $35 per month and a call charge of $0.40 per minute

To make the problem simpler, if only a part of minute is used, only that part of a minute is charged to the user.

Think about the needs of new customers of OZ-Mobile. Write a brief report explaining which scheme will be best for customers, based on the number of minutes that they might be on-line each month. Make sure you indicate when each of the schemes is most cost-efficient.

(Some parts of question omitted here due to space restrictions)

Figure 1 Mobile Phones Problem (adapted from TEA materials, in press)

Range, scale or a visual approach.

Both Graphmatica and the TI-82 have simple ways in which the dimensions of the viewing window, the 'range' of x and y values in the viewing rectangle, can be specified. Because of their ease of use and obvious utility, these features are used frequently to arrange for the relevant part of the graph to appear in the viewing window. Our TI-82 user, in particular, tended to set the range for the viewing window rather than locating the relevant part by zooming out or by setting the scale. To do this, he developed routine procedures beginning with estimation and then using a special aspect of the trace function . In the Mobile Phone problem (Fig. 1), for example, he began to find the right viewing window by observing (from answers to intermediate questions omitted or from the fact that at a break-even point the extra $10 per month rental for Scheme B will need to be compensated for by saving 40 cents per minute) that the points of intersection are likely to be near x = 25. He entered and plotted the three functions and then set the range so that the x values include x = 25 (with any y-values). Then he moved the cursor to any point with x = 25 and used a special feature of the "trace" function which automatically locates the graph of y_1 and moves to a new viewing window centred around the point on the graph with x co-ordinate 25 (see Fig. 2). He finetuned the viewing window as required and used the trace function to read off the points of intersection of the graphs. His thinking centred around range and he sought rough estimates of the numerical values of likely solutions.

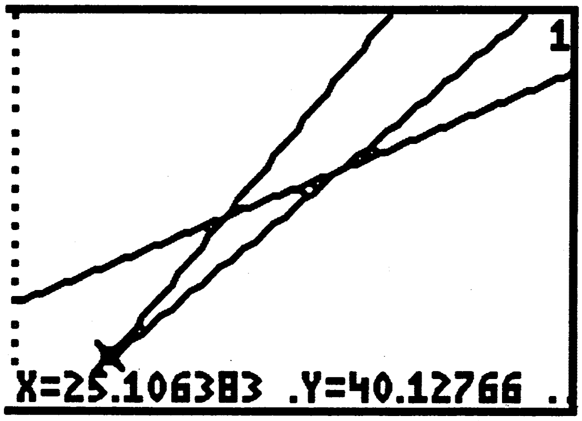


Figure 2 A graph of the Mobile Phones problem (using the TI-82).

Our Graphmatica user rarely thought about range, but developed efficient techniques using complex combinations of zooming out and zoom-box. By using a very narrow zoom-box between standard zooming out moves, the Graphmatica user was able to attain the very different scales on the axes that are often required to obtain a good viewing window. For example, to show a reasonable picture of Stella's pay options for the Movie Problem (Fig. 3), the default window would have to be deformed into one with values of x from about 14 to 21 and values of y between about 30000 and 70000000. The zoom-box is a highly visual and intuitive approach to adjusting the range and the scales. We noticed that the Graphmatica user did not think in terms of either the range or the scale, but instead worked with notions of getting a closer or a more distant view.

In contrast to the other two function grapher users, the ANUGraph user routinely thought about scale. ANUGraph (V2.08) has an extremely cumbersome method for defining the range explicitly. To do this requires using menu options for stretching or shrinking an individual axis. The ANUGraph user thinks about scale on the axes in ways which are not required with the other two function graphers. Despite the fact that issues of scale underlie all changes to the viewing window, users of Graphmatica and the TI-82 never have to confront it. We know that choosing the scale is one of the main difficulties that students encounter when drawing graphs by hand to fit onto a given piece of paper (Barcham, 1994). We now suspect that this will be one idea that is not assisted by the use of function graphers as visual intuitive methods become standard features.

The Movie Contract

Stella Rosengren, the movie star of the century, has recently been offered some unusual contract inducements. One studio offered her a choice of three salary options. Shooting takes between 16 and 20 days.

Option A $20 for the first day of work, but her overall pay doubles for each additional full day of work.

Option B Two cents for the first day of work, but her overall pay triples for each additional full day of work.

Option C A flat rate of $100,000 per day for as many full days as the movie is being shot.

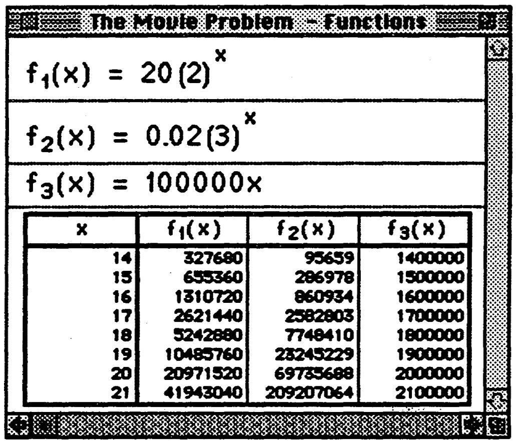
Write a report to Stella, outlining your considered advice about the best option to take.

(Some parts of the question omitted here due to space restrictions)

Figure 3 Movie Contract Problem (adapted from [Phillip, 1993])

Tables of values

We noted marked differences in the frequency with which we used tables of values when solving the problems with different function graphers. ANUGraph (V2.08) has an excellent tables feature, which our user relied upon heavily. The user can make tables where the values of several functions appear side by side to make comparisons. Thus, for example, the amounts of money earned by Stella (see Fig. 4) can be readily printed out for 14 -21 days and the problem can be quickly solved.


The Movie Problem - Functions

$f_1(x) = 20(2)^x$

$f_2(x) = 0.02(3)^x$

$f_3(x) = 100000x$

x	$f_1(x)$	$f_2(x)$	$f_3(x)$
14	327680	95659	1400000
15	655360	286978	1500000
16	1310720	860934	1600000
17	2621440	2582803	1700000
18	5242880	7748410	1800000
19	10485760	23245229	1900000
20	20971520	69735688	2000000
21	41943040	209207064	2100000

Figure 4 Using ANUGraph's List Values feature to solve the Movie Problem.

On the other hand, Graphmatica (in the current version V1.3c) does not create tables very flexibly. The user cannot control the value of x at which the table starts nor the size of the step of the tables and the values of only one function can be seen at a time. Our Graphmatica user rarely used tables, preferring to use the very flexible zooming in and out. This routine persisted even in the Movie Problem, which illustrates just how unproductive the process of zooming out could be (see Fig. 5). Continually zooming out would not make the linear graph f(n) = 100,000n appear at all.

Standard viewing window After zooming out four times

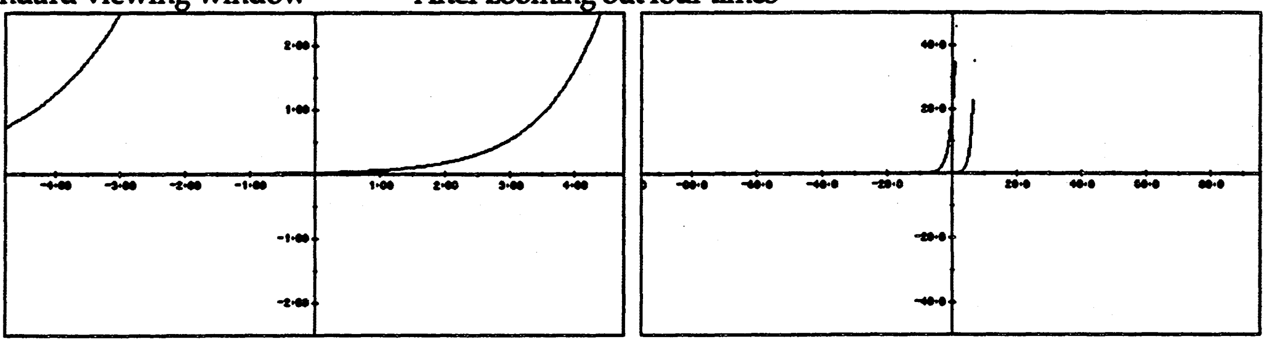


Figure 5 Using zoom out unproductively with the Movie Problem.

Recursively defined functions

Of the function graphers we used, only the TI-82 could define functions recursively, as is now commonly done with spreadsheets. For example, when a spreadsheet is used for the Movie Problem, tables of values may be created recursively, such as $C_{n+1} = 3\ C_n$, $C_1 = 0.02$, rather than by first deriving an explicit function such as $C(n) = 0.02 * 3^{n-1}$. The problem statement defines the pay under each option recursively as a function of time and a spreadsheet allows these to be directly programmed. The resulting table can then be used to examine the relative merits of each salary option in the relevant period. Although the chart produced by a spreadsheet (see for example Fig. 6) is not as precise or flexible as that produced by a function grapher, it does give a good indication of the overall situation and has automatically adjusted the scaling so that the calculated values fit into the viewing window. This is an example of the blurring between function graphers and spreadsheets which we predict will continue and further multiply the variety of methods people use.

n	A(n)	B(n)	C(n)
1	100000	20	0.02
2	200000	40	0.06
3	300000	80	0.18
4	400000	160	0.54
5	500000	320	1.62
6	600000	640	4.86
7	700000	1280	14.58
8	800000	2560	43.74
9	900000	5120	131.22
10	1000000	10240	393.66
11	1100000	20480	1180.98
12	1200000	40960	3542.94
13	1300000	81920	10628.82
14	1400000	163840	31886.46
15	1500000	327680	95659.38
16	1600000	655360	286978.14
17	1700000	1310720	860934.42
18	1800000	2621440	2582803.26
19	1900000	5242880	7748409.78
20	2000000	10485760	23245229.34
21	2100000	20971520	69735688.02

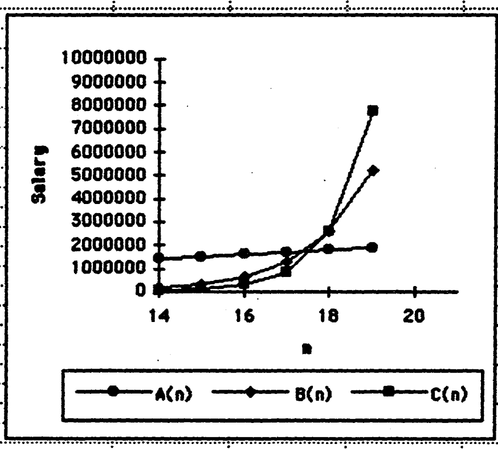


Figure 6 A spreadsheet solution to the Movie Problem.

Conclusion

Although these and other function graphers share the same basic facilities, their additional capabilities are varied and the ways in which different tasks are to be done vary in difficulty. Each has some features which become preferred and others which are avoided. We believe that they are all excellent function graphers and that any of them are excellent choices for schools. However, they make different things easy and this results in users developing different approaches to problems. Further, the decisions software developers make about notation and selective emphasis on capabilities will to some extent influence the patterns of thinking that students develop. In summary, we have found that the use of software (such as function graphers) changes more about the practice of mathematics and its teaching than we had expected.

References

Asp, G., Dowsey, J., & Stacey, K. (1992). Technology enriched instruction in year 9 algebra. In Southwell, Perry, & Owens (Eds.), Fifteenth Annual Conference of the Mathematics Education Research Group of Australasia., (pp. 84-93). Nepean: MERGA.

Asp, G., Dowsey, J., & Stacey, K. (1993). Linear and quadratic graphs with the aid of technology. In Atweh, Kanes, Carss, & Booker (Eds.), Sixteenth Annual Conference of the Mathematics Education Research Group of Australasia., (pp. 51-56). Brisbane: MERGA.

Barcham, P. (1994). Student interpretation of graphs at secondary level. Unpublished M.Ed. thesis, University of Melbourne.

Dubinsky, E. (1994). A Theory and Practice of Learning College Mathematics. In Schoenfeld (Ed.), Mathematical thinking and problem solving (pp. 221-243). Hillsdale: Lawrence Erlbaum Associates.

Phillip, R., Martin, W., & Richgels, G. (1993). Curricular implications of graphical representations of functions. In Fennema, Romberg & Carpenter (Eds.), Integrating research on the graphical representation of function (pp. 239-277). Hillsdale: Lawrence Erlbaum Associates.

Steele, D. (1995). The Wesley College technology enriched graphing project Unpublished M.Ed. thesis, University of Melbourne.