Graphical Packages for Mathematics Teaching & Learning

David Tall

Mathematics Education Research Centre
University of Warwick
COVENTRY CV4 7AL
England

The term “graphical package” is of very recent vintage, and yet is already proving inadequate to describe the wide variety of versatile software using high resolution computer graphics. The name arose at a time when text was the norm and expensive graphics facilities were a luxury to which few had access. A graphical package was then an additional facility to draw pictures to insert into word-processed text or to represent information pictorially. Now that all modern micros include high resolution graphics, virtually every piece of software that is not purely text-oriented could lay some claim to the title.

In this article we will restrict our discussion to those graphical packages having a direct bearing on the teaching and learning of mathematics. These include general packages for illustrating numerical work carried out in spreadsheets and data-bases. Of greater interest for our purpose are those to demonstrate and explore concepts in many areas of the mathematics curriculum, including geometry, probability, statistics, calculus, differential equations, mathematical modelling, and even less obviously graphical topics, such as algebra, which benefit substantially from being related to graphical ideas. It is no accident that when we understand something we often say that we “see” it. The visual side of the learning process is one that complements the purely verbal, and flexible graphical packages provide us with very powerful tools to enhance this much-neglected aspect.

1. GRAPHICAL PACKAGES

Graphical packages appear in all kinds of variety and levels of sophistication. A recent development is the appearance of hand-calculators with liquid crystal displays to give coarse representations of the graphs. The Casio fx-7000G, for example, will draw graphs of functions, bar-graphs and line graphs on a 95 by 63 grid of pixels, giving a reasonably satisfying picture; future technology will no doubt soon radically increase the quality of the display. It is packed with clever ideas to help the user. For instance, if the variable is omitted in a function, the calculator uses sensible default values for the range. Thus the graph of SIN is from -2π to 2π, whilst the graph of SINx will be superimposed on

Published in *Informatics and the Teaching of Mathematics*,
(ed. Johnson D.C. & Lovis F.), North Holland, pp. 39–47, 1987.

the current picture. Despite such subtle facilities, or perhaps because of them, the system lacks “user-friendliness”, betraying its origins as an updated calculator, with multiple functions on every key requiring familiarity and practice to cope with their possible usage.

An innovative idea, proposed to improve the students “visual and qualitative feel for the properties of mathematics” is the “computer illustrated text” (Harding [1]). Here “normal figures and diagrams are replaced by screen displays which can be produced dynamically by the reader, offering a far greater variety of illustrations than would be possible with printed diagrams alone.” The student may read the book in the normal way, turning to the computer to see the dynamic illustrations, but may also actively participate in the learning process by exploring the concepts with the aid of the software.

The programs in a package such as this are often intentionally unsophisticated, designed to focus on the essential ideas without encrusting them with error-trapping routines and flexible facilities. Major graphical packages are usually carefully error-protected and designed to encourage the user to explore the concepts far more freely.

The *Geometric Supposer* [2] enables the user to make geometrical constructions which may then be repeated on other figures, to see whether the consequences of a given construction are generalizable. The intention is to transform a student into “potent and nimble conjecture-maker”. The user is given appropriate menus of mathematical tasks from which to choose, each choice requiring only a single key-stroke, Thereafter it is necessary to type in mathematical commands, such as those in the following diagram to measure the ratios of sides in similar triangles. (Figure 1.) The interface is designed to encourage the linking of each geometrical concept with corresponding mathematical symbolism.

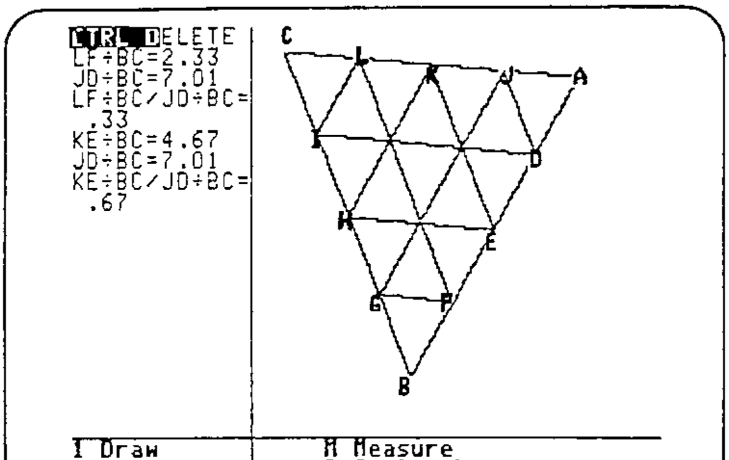

Figure 1 : The Geometric Supposer

Graphic Calculus [3] is a suite of programs using numerical techniques to explore graphical representations of calculus concepts. It begins by magnifying graphs to discover that many are “locally straight” under high magnification. This leads to the notion of the gradient of a curved graph, carried out numerically by the computer and visualized at each point by a process of mental magnification. The purpose of this approach is to give a geometric idea of calculus concepts to relate to the usual algorithms of differentiation and integration, and to the solution of differential equations.

MacSpin [4] enables the user to explore statistics graphically. For example, up to ten sets of statistics can be input and analysed by selecting any three, plotting a scattergram in three dimensions and allowing the user to view the picture from any angle to see if there is a visible correlation. The rotation is performed smoothly by selecting onscreen icons indicating the required rotational axis and gives the user the clear illusion of a collection of points being rotated in space. (It is, however, very difficult to represent such an illusion in a static picture in a text, such as in figure 2.) This experience gives intuitive support to what was previously just a number-crunching exercise in factor analysis.

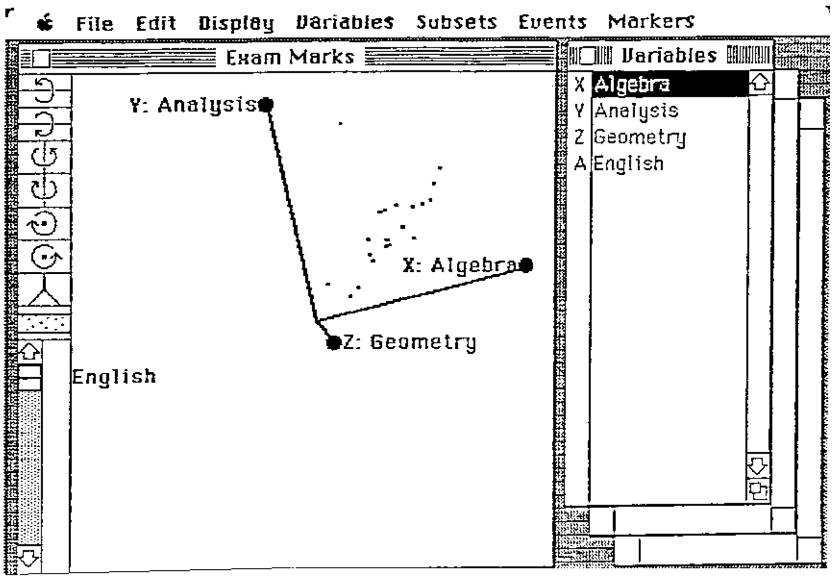

Figure 2 : MacSpin

Phasar [5] is an sophisticated interactive simulator for difference and differential equations written to

> “provide the necessary tools for live demonstrations during lectures, and for experimentation and research by students and faculty in applied mathematics.”

A large number of options are available with single keystrokes from onscreen menus, and starting positions may be saved to disc, enabling them to be replayed at a later date. For example, it is possible to study solutions of differential equations in three dimensions, taking advantage of choice of rotation and perspective projection, and the system can handle equations in four dimensions by selecting three of the four variables for viewing and using the three-dimensional routines. (Figure 3.)

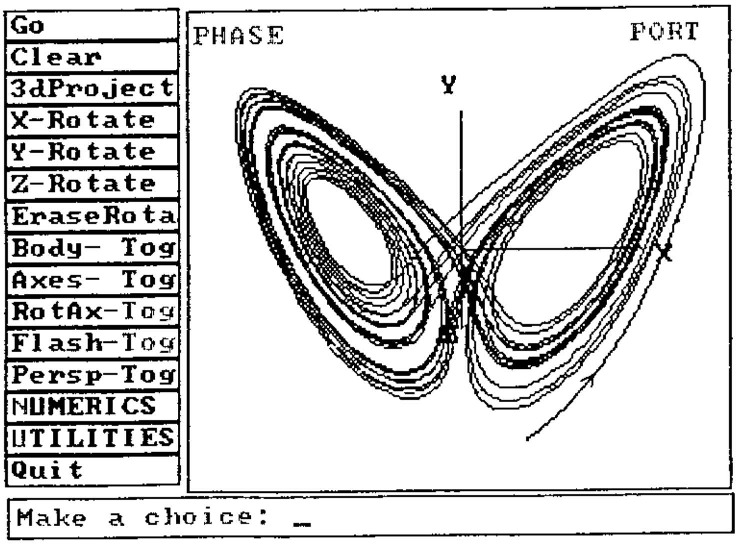

Figure 3 : Phasar

Stella [6] is a dynamic modelling system, enabling the user to set up relationships between variables as formulae, logical statements, or piece-wise defined graphs which can be set up by moving the shape of the graph around onscreen. It can then be set in motion to see what happens to the variables during the passage of time.

All these packages encourage intuitive exploration and this is a regular theme recurring in other graphical software:

> "Our object is to give a 'feel' for probability distributions rather than go into details of the mathematics or special applications."
> (Robinson & Bowman, *Introduction to Probability* [7])

> "There are many investigations, allowing one or more learners to work together, or with a teacher, to develop a mental picture of the ideas to supplement the formal manipulations."
> (Tall, *Graphic Calculus*[3])

> "The real purpose of the program is to help the students expand their intuitions and their capacity to deal with abstractions."
> (Schwartz & Yerushalmy, *The Geometric Supposer* [1])

The reason for such a unity of purpose in so diverse a set of software packages is part of a deeper cognitive need in human learning.

3. INTUITION AND RIGOUR

In the early years of the twentieth century, Poincaré [8] made an important distinction between two types of mathematician:

> "The one sort are above all preoccupied by logic; to read their works, one is tempted to believe they have advanced only step by step, after the manner of a Vauban who pushes on his trenches against the place

> besieged, leaving nothing to chance. The other sort are guided by intuition and at the first stroke, make quick but sometimes precarious conquests, like bold cavalrymen of the advance guard."

This distinction between the logical and the intuitive has often fascinated the mathematical psychologist. There are elements of both in all of us and it is the balance between these two complementary modes of operation that is of central importance in mathematical education.

In the early seventies, long before graphical packages became a reality, Skemp contrasted the roles of visual and algebraic symbolism: the former he saw as integrative, holistic, capable of simultaneous representation of ideas, the latter analytic, detailed, logical and sequential. He also put forward the hypothesis that verbal communication was a social phenomenon in which all present (including the speaker) could hear the words being spoken, whereas visual imagery was more personal and less easily communicated:

> "We have built-in loudspeakers, but not built-in projectors."
>
> (Skemp [9])

Graphical packages, if suitably flexible, become tools for man to communicate in pictures. They provide an opportunity for a new approach to learning in general and mathematical education in particular. For this to happen, the software needs to be flexible and powerful under the control of the user.

4. THE USER INTERFACE

Two different kinds of approach seem to be implicit in software, one expecting a technical decision, the other offering a more intuitive mode of operation. Earlier computer operating systems offered only the former, forcing the user to be aware of the technicalities of the machine. More recent operating systems, using a Windows/Icons/Menus/Pointer environment, allow the user to make a choice simply by pointing to a box onscreen or moving the graphic representations in an intuitive manner using a "mouse".

The two modes of operation are often linked to the cognitive manner in which the software is used, and cause differences of approach. For instance, Harding & Quinney [10] do not provide the user with the option of automatic scaling of graphs in their programs to calculate the numerical solutions of equations, on the grounds that "setting the *y*-range by hand often helps to focus the user's attention on the values". This causes a process of logical reflection on the initial picture. Automatic re-scaling, on the other hand, encourages intuitive exploration of the limiting process itself. For instance, the initial picture may show the desired solution as the intersection of a curved graph with the *x*-axis, but high magnification may show a tiny portion of the curved graph as being approximately straight, reducing the problem essentially to that of the intersection of two straight lines (figure 4).

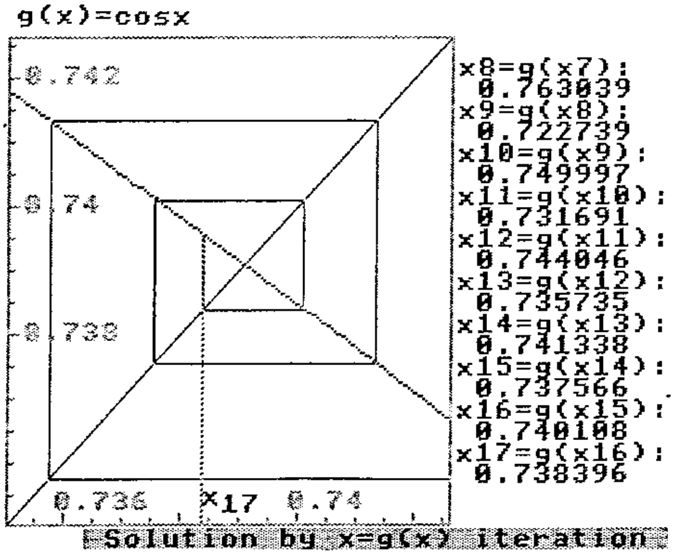

Figure 4 : Zooming in on the solution process

The two styles of operation are also related to different modes of information-processing, the first often quantitative, offering great precision, the second qualitative and more open to visual clues. For example, the program Phasar, to draw solutions of differential equations, allows two solutions to be drawn simultaneously which begin at almost the same place, yet suddenly diverge at a later stage. Without a high precision input, this phenomenon would be impossible to demonstrate.

On the other hand, if one has a picture onscreen and wishes to look at it from a different angle, it may be preferable to select the view-point using a pointer. For instance the *3D Grapher*, [11], draws a picture of a surface and the viewpoint may be changed by dragging an “eye” around in a window representing the horizontal plane of the domain, or it may be raised or lowered by shifting an “eye level” marker. As the eye is dragged around, the surface is constantly being redrawn in rough detail, until the chosen viewpoint is selected and the drawing is done with high-level accuracy (figure 5).

As programming becomes more sophisticated, a combination of both methods is often available, usually in the form of an intuitive selection using a mouse or other device to make a precise technical input. For example, parameters represented symbolically or numerically onscreen, might be selected using a pointer, then edited using input from the keyboard.

In general, a good graphical package should retain the benefits of both types of thinking: technical decisions for sequential, logical thought and intuitive operation for global, analogical insight.

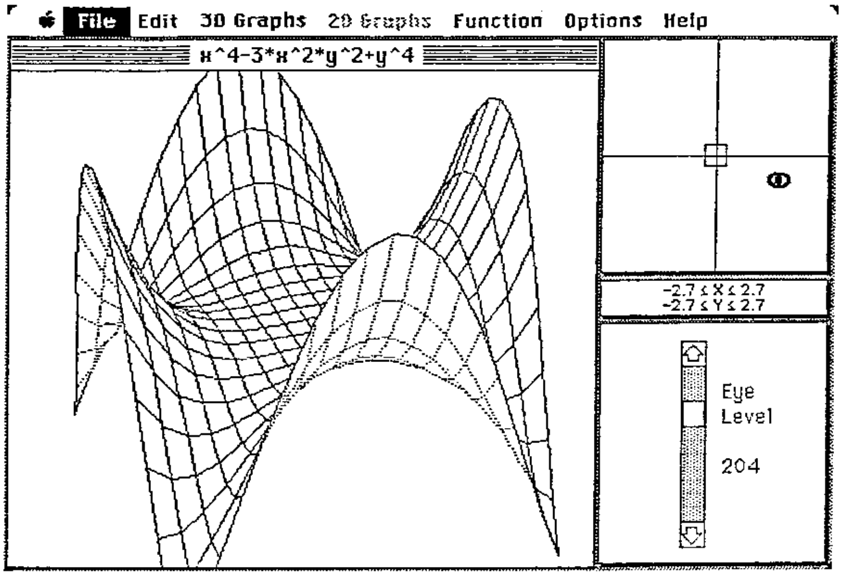

Figure 5 : 3D Grapher

5. EXTENSIBILITY AND PROGRAMMING

Most packages described so far require no programming, the exceptions being the less sophisticated programs in some computer illustrated texts. The idea is to encourage students to modify the code, should they wish to carry out a task slightly different from the one envisaged. In later texts in the series the policy was occasionally abandoned:

> "Numerical methods often require complicated setting of data and ideally need both graphical and numerical results to be displayed. ... These considerations tend to require quite elaborate programs and we felt that it was not practicable to make them easily readable, especially for machines with limited internal storage." ([10], page vii.)

Graphic Calculus started out with the opposite view of producing software without the need for any programming but the possible misunderstandings that occur when one attempts to model continuous calculus by discrete numerical methods led to a reappraisal and the introduction of short programs to illustrate the algorithms. Koçak is only too aware of the problem in his software for dynamical systems:

"The danger of oversimplification is quite real, and it is easy to get the false impression that computers can provide the answers to all questions. The user should always remain aware that numerical simulations have the potential to be misleading." ([5], preface.)

The language Logo has long been used in the classroom for pupils to produce geometrical shapes by their own efforts in programming. In a way the Logo language is itself an archetypal graphical package (though it is also much more than that). To help students use Logo in more complex situations, Papert and others has suggested the use of "microworlds" which consist of a set of procedural facilities enabling the user to carry out appropriate tasks in a given context. Such systems have the advantage that they are “extensible” and modifiable by the user to suit related needs. In general the other graphical packages so far described are closed, inextensible systems, albeit very flexible ones. Indeed, to be able to operate in the intuitive mode that encourages free exploration, the logical activity of programming may not always be appropriate.

6. MULTI-REPRESENTATIONAL SOFTWARE

All of the packages so far described have more than just a graphical display. They usually enable graphical interpretation of other kinds of data. Some software is explicitly designed to exhibit the relationship between different methods of data representation. For instance, the Semantic Calculator, now under development by Judah Schwartz at the Educational Technology Center, Boston, encourages the user to take information given verbally, to translate it into algebraic form, and then to represent it graphically.

Multi-representational software is especially valuable when it allows the translation of precise numerical or symbolic information into graphical form. This is happening in much professional business software, where numerical data is translated into bar-charts or pie diagrams, and it is at its most powerful when the data and its visual representation may be manipulated at will to see what happens when certain parameters are changed.

Symbolic software,originally designed on large main frame computers in Lisp and related languages initially had no graphics available. The symbolic manipulator *muMath* [12] now has a prototype offering graphic representation of the data. *Powermath* [13], one of the early symbolic manipulators on the Macintosh was published with the possibility of a graphical display, but the graphics proved to be so rudimentary that they were almost an embarrassment.

New languages, such as *Turbo Prolog*, offer a distinct possibility of uniting symbolic manipulation with speed of calculation and graphical display. Writing in this language, Jens Ole Bach of Roskilde University Denmark has already produced fast operating graph drawing packages that include routines to demonstrate and carry out symbolic differentiation.

7. NEW DIRECTIONS FOR THE MATHEMATICS CURRICULUM

Graphical packages present new opportunities in the mathematics curriculum. They allow new concepts to be approached visually and intuitively before the need for logical analysis. The logical ideas themselves, and the links to the visual representations still need to be established, but they can now be performed in a context where the student has some overall idea as to what the new concept represents.

In my thesis [14], the notion of gradient of a graph was approached in this way, with a parallel study of the usual formal algorithms. Compared with a control group, the students with graphical experience had far greater facility in sketching the gradients of given graphs (though this had not been taught explicitly), in recognizing the graphs of gradient functions and in giving examples of non-differentiable functions, without any adverse effects on their ability to carry out the standard algorithms of differentiation.

A valuable use of many of the graphical packages is a form of mathematical creativity developing out of exploration in a rich conceptual environment. In the introduction to the *Geometric Supposer*, Schwartz & Yerushalmy comment on the lack of provision for students to play an active and generative role in learning mathematics.

> "We don't teach language that way. If we did, we would never require students to write an original piece of prose or poetry. We would simply require them to recognise, appreciate, and memorize the great pieces of language of the past, literary equivalents of the Pythagorean Theorem and the Law of Cosines."

Schwartz has also cautioned against premature enthusiasm, saying "its terribly important not to be deluded by the excitement of the greenhouse." But experience in the classroom is showing the potency of these exploratory tools for gaining insight into sophisticated mathematical concepts.

At the research level too it is instructive to see graphical packages used for exploration to generate conjectures. The program Phasar may be used to draw known phenomena in dynamical systems for which formal proofs are still awaited. Indeed, the default equation in the library of three dimensional differential equations is the Lorenz attractor, whose behaviour "could not have been discovered without the computer". ([5], page 138.)

Thus Graphical Packages have a vital role to play in future mathematics, expanding the vision not only of young pupils, but also the frontiers of mathematical research.

REFERENCES

[1] Harding R., *Fourier Series and Transforms, A Computer Illustrated Text* (for the BBC and Apple Computers), (Adam Hilger, Bristol & Boston, 1985).

[2] Schwartz J. and Yerushalmy M., *The Geometric Supposer: Triangles* (for the Apple Computer), (Sunburst Communications, Pleasantville, N.Y., 1985).

[3] Tall D.O., *Graphic Calculus I,II,III* (for the BBC Computer), (Glentop Press, London, 1985).

[4] Donoho A.W., *MacSpin* (for the Macintosh Computer), (D^2 Software Inc., Austin, Texas, 1985).
[5] Koçak H., *Differential and Difference Equations through Computer Experiments* (for the I.B.M. Computer), (Springer-Verlag, 1986).
[6] *Stella*: High Performance Systems, 13 Dartmouth College Highway, Lyme NH 03768).
[7] Robinson D.A. and Bowman A.W., *Introduction to Probability*, (for the BBC and I.B.M computers), (Adam Hilger, Bristol & Boston, 1986).
[8] Poincaré H., *Mathematical Creation*, in: Halsted G.B., *The Foundations of Science*, (The Science Press, New York, 1913).
[9] Skemp R.R., *The Psychology of Learning Mathematics* (Pelican, London, 1971).
[10] Harding R. and Quinney D.A., *A Simple Introduction to Numerical Analysis*, (Adam Hilger, Bristol & Boston, 1986).
[11] Lewis H., Matthews J. and Kuperberg G., Math21: *3D Grapher* (for the Macintosh Computer), (Harvard University, 1985).
[12] Stoutemyer D.R., *muMath* (for the I.B.M. & computers running CP/M), (The Soft Warehouse, Honolulu, Hawaii, 96822).
[13] Roth C.E. and Davenport J.H., *Powermath* (for the Macintosh Computer), (Industrial Computations Inc., Calabas, California, 1985).
[14] Tall D.O., *Building and Testing an Approach to the Calculus using Interactive Computer Graphics*,(Ph.D. thesis, Warwick University, 1986).