Global J. of Engng. Educ., Vol.7, No.1
Published in Australia
© 2003 UICEE

Tackling the Mathematics Problem with *MathinSite**

Peter Edwards
Paul M. Edwards
Bournemouth University
12 Christchurch Road, Bournemouth, BH1 3NA, England, United Kingdom

Mathematics is the stumbling block for many engineering students worldwide, often due to a lack of a *feel* for the subject. In order not to disenfranchise and subsequently fail otherwise capable students, every effort has to be made to enhance their feel for, and hence understanding of, mathematics. A project is aimed at deepening student mathematical understanding and is funded from Learning and Teaching Development initiative of Bournemouth University, Bournemouth, England, UK, and the bursary accompanying a UK National Teaching Fellowship [1]. This project has resulted in the translation of some *Visual Basic* mathematics programs into Java applets (small programs that run through Internet browsers such as *Internet Explorer* and *Netscape*) delivered on the Internet Web site, *MathinSite*. This article presents an investigation into some of the causes and repercussions of the so-called *mathematics problem* including the need for projects such as *MathinSite*. *MathinSite*'s philosophy is examined, giving some reasons for choices made and an outline given of some of the problems and solutions encountered. The experiences described here could prove useful to others planning similar ventures.

BACKGROUND

In a keynote address to the assembled delegates at the *2002 Annual Conference of the American Society of Engineering Educators*, held in Montreal, Canada, Dr William A. Wulf, President of the American National Academy of Engineering, said of the student retention rate on engineering courses:

> *It is a disgrace! Depending on whose numbers you use, something approaching half the students entering engineering do not finish with an engineering degree. Those who leave are not poor students; by almost every measure they are about the same calibre as those who stay. We are not* weeding out *the poor students!* [1].

While he goes on to say that *the poor retention rate is a measure of our failure to convey the pleasure and impact of engineering*, there is no doubting that this appalling figure is considerably higher than, for example, undergraduate courses with little or no analytical content. So, while there may be many personal reasons behind student withdrawal from engineering courses, the finger of suspicion certainly points towards learning and teaching deficiencies in mathematical subjects as the prime culprit of poor student retention. Assuming this to be the case, it becomes necessary to investigate why students' skills in mathematics have declined, why they are not engaging with mathematics in the first place and what can be done to alleviate the problem.

THE MATHEMATICS PROBLEM

The general decline in undergraduate mathematical ability (the *mathematics problem*) was already being highlighted in a number of reports written as long ago as the mid-1990s [2-4]. At present, this general decline shows little sign of improving.

There are a variety of reasons that have contributed to this problem - some measurable and some apocryphal. Among the measurable factors are:

*A revised and expanded version of a lead paper presented at the 3rd *Global Congress on Engineering Education*, held in Glasgow, Scotland, UK, from 30 June to 5 July 2002. This paper was awarded the UICEE silver award (joint fourth grade with one other paper) by popular vote of Congress participants for the most significant contribution to the field of engineering education.

- During the 1970s, 1980s and into the 1990s, there was a shift in early school mathematics away from the skill of accruing a mathematical knowledge base to the more pupil-friendly *investigative mathematics*. This included, among other things, a decline in the rote learning of multiplication tables. The overall effect was to deprive pupils of having basic mathematical knowledge and skills *at their fingertips*.
- The advent of the electronic calculator compounded the above problem in that pupils would slavishly believe what their calculators showed and did not have the wherewithal in terms of basic numeracy to validate their answers. Although not true for all, many lacked a feel for numbers.
- With a reduction in skills at the lower school level, it was to be no surprise that the number of students undertaking optional higher-level school mathematics courses (from the age of 16 to 18) would show a continuous decline. Compounded with this has been the proliferation of alternative courses that students can study in their final school years that contain little or no mathematical content – easier options for students whose mathematical skills in their earlier school lives had already been left wanting.
- Engineering courses need students and, in particular, engineering courses need mathematically able students. With a reduced quality of mathematical learning in early school and, for some, the complete avoidance of mathematics between 16 and 18 years old, engineering courses have been forced to take on already weak students with, further, a two-year hiatus in their mathematical studies. With widening access, this hiatus could be 10 or more years for mature students.

Mature students (who may otherwise offer extensive and valuable acquired previous experiential learning) can often be accommodated by the many undergraduate engineering courses now offering bridging or foundation courses to assist those with large gaps in their learning experience – especially in mathematics.

For direct entrants from school, the longer-term future, fortunately, looks brighter. In the last few years, certainly within the UK, early school experiences in mathematics have been revised. Some of the *fun* side of mathematics (the investigative mathematics) has been retained, even though it is time-expensive, but at the same time, there has been a reintroduction of the need for pupils to accrue basic mathematical knowledge and skills. This has included the decision to exclude calculators in the early school years. With a better grasp of basic numeracy and early-school mathematics, and a greater confidence in their own mathematical ability, the hope for the future is that more students will take on higher level mathematics - and that those who do not will at least be able to offer a better understanding and knowledge of basic mathematical skills. However, in the meantime, there are still school-leavers coming through the system who have problems with mathematics as they commence engineering courses.

WHAT ARE THE PROBLEMS AND SOLUTIONS?

Some mathematical misconceptions and a lack of background knowledge were discussed in a paper investigating results from a mathematics diagnostic test given to 208 first-year undergraduate design engineers [5]. For example, nearly 75% of one cohort indicated that they had no feel for the meaning of m and c in $y = mx + c$. Many were unable to add two simple fractions - most attributing this to the (over)use of calculators in school at the time when fractions and similar calculations were introduced.

In trying to cope with these problems in undergraduate engineering courses, it is important neither to lower standards by diluting analytical content nor to change the emphasis of existing courses. However, a university would not want to turn away prospective students who may well excel in all required areas other than mathematics. In order to accommodate such students, mathematics diagnostic testing is often used during the first week of a student's undergraduate course. However, diagnostic testing alone is insufficient; comprehensive follow-up support must be provided.

Quite often, follow-up support takes the form of *extra maths* classes, *drop-in centres* or *guided reading* lists. These undeniably play a useful role in helping mathematically less able students, but there is no doubt that computers and calculators can also play a part. During the 1990s, computer algebra packages, such as *Maple* and *Mathcad*, and software tutorial packages such as *Mathwise*, became widely available [6]. Even spreadsheet packages were recognised as a valuable tool in aiding mathematical understanding. However, writing and maintaining the integrity of the user files can be time-consuming and hosting the required software packages can be expensive.

Graphical calculators are excellent in giving instant visual feedback (although it may be unfair to expect all students to have to purchase them). Not only can graphical visualisations open up a new world to those students for whom mathematics is seen to consist only of formula manipulation, but they can also aid students for whom memory recall can be difficult.

Remembering mathematical ideas and techniques can be a problem. Students will depend heavily upon *semantic* memory, for example, under examination conditions where careful and accurate recall is required. Semantic memory relates to the meanings of things, such as words or concepts. In contrast, *episodic* memory is the remembering of events, things that happened, places visited, based upon sequences of events. Calling on episodic memory requires event reconstruction (*…and then I did this…*), which, unfortunately on the one hand, can be susceptible to an individual's perceptions and interpretations but on the other hand, can complement semantic memory. Using interactive visualisations can aid memory stimulation and retention.

Visualisations can also add vitality and meaning to the mathematics. The very act of producing visualisations through a sequence of events and then encouraging students to sketch the visualisation's graphs from the screen can help to draw the student in to the subject - and thereby promote reflective learning.

In summary then, structured dynamic graphical visualisations can play an important part in enhancing the learning process by intensifying student engagement, improving memory retention and deepening understanding.

THE *MATHINSITE* PHILOSOPHY

With the foregoing in mind, when the first author proposed *MathinSite* as a way of helping to alleviate the mathematics problem of his own students, the learning-by-doing graphical approach to be provided by *MathinSite*'s dynamic, interactive visualisations was set as the cornerstone of its learning and teaching strategy [7].

But what was the best way of making these visualisations available?

With the increasing sophistication of (free) Web browsers and the introduction of platform-independent Web programming languages such as Java, providing students with readily available graphical mathematics support over the Web has become a reality without having to resort to purchasing proprietary software packages or individual graphical handheld calculators. It was decided, therefore, that the software support offered by *MathinSite* would be in the form of Java applets (applet = small application).

However, graphical applets alone would not necessarily enable students to discover important mathematical points on their own. Consequently, it was decided that an integral part of the overall package would be to include a set of theory and tutorial worksheets accompanying each applet; these were to lay prescribed paths through each applet's use. This *guided tour* would not only help the first-time user of an applet (unsure of what to do or what to see), but also help students feel as though they were discovering aspects of mathematics for themselves, aspects that would not be readily apparent from formulae, equations or graphs in a book. Furthermore, being designed specifically as an integrated package, all of this could be done as a *supplement* to, and not as a *replacement* for, normal class-contact time.

As already mentioned, insight can be achieved through visualisation – but a deeper insight can be achieved through *dynamic* graphical visualisation.

Even with an accompanying *static* plot (itself a visualisation), the mathematical statement $y = A\sin(\omega t + \varphi)$ has little meaning to some students. However, software that allows students to change A, ω and φ dynamically, so changing the plot in real-time, can provide a deeper understanding of how the equation works. On its own, such software does not necessarily promote deeper insight. But with the tutorial worksheets asking *the right questions* and requiring students to make pertinent notes and drawings from the screen, this becomes an invaluable aid in promoting mathematical understanding and in enhancing the students' learning experience.

Students are able to further test their understanding with the tutorial exercises at the end of each worksheet. No answers are given to these tutorial exercises. If working with the applet has not already given students the insight to answer the exercises successfully on their own, then further use of the applet will help them find the answers!

A visualisation on its own cannot answer all the questions, especially when it comes to underlying mathematical theory. This can only be achieved from the theory sheets (or, in some cases, a small amount of theory contained within the tutorial sheets). How the theory sheet is used is up to the student (or supervising lecturer). Some may have already covered the theory; others may like to see the applet in action before investigating the underlying concepts.

Using *MathinSite* in lectures and tutorials can prove invaluable. As a simple example, instead of having to draw several straight lines and their associated equations on the board, the lecturer can easily demonstrate dynamically how changes in m and c affect the straight line. If necessary, students can then reinforce this by trying the applet themselves in their own time. Students do not need to have seen an applet in use in class. Not only is each applet/theory/tutorial sheet combination self-contained, but also, with all applets and accompanying worksheets available on the Internet, access can be made from any Internet-ready computer at any time of day (or night), so any of the applets can be explored at leisure. Using the applets on their own, students can take ownership of their own learning; learning that can be undertaken at *their* own pace in their own non-threatening environment. Ease of use is assured with a consistent user-interface across all applets and user-instructions in the accompanying worksheets. Failing that, *MathinSite* also provides Lotus *ScreenCam* movies showing how to operate the software.

The philosophy outlined here was accepted at the authors' University as an important way forward in enhancing student learning, notwithstanding the project's proposal stating that there would not necessarily be a *direct* saving in lecturer costs. It was not seen as a means of reducing teaching contact hours but more a means of saving money by reducing student attrition due to failure in analytical subjects.

This project is aimed at deepening student mathematical understanding and is funded from Learning and Teaching Development initiative of Bournemouth University, Bournemouth, England, UK, and the bursary accompanying a UK National Teaching Fellowship [8].

CHOICE OF MATHEMATICS AND INPUT MODE

Basic functions with meaningful graphical representations were the obvious starting point. These included:

- The straight line, $y = mx + c$ (not *too* basic for such an undertaking since, as already described above, not all students are familiar with the meaning of the parameters m and c).
- The parabola, $y = ax^2 + bx + c$ (students may be able to describe the effect of changing a and c, but how many can describe the effect of changing b?).
- Functions such as $y = A\sin(\omega t + \varphi)$ and $y = A + Be^{kt}$ (fundamental functions in engineering).

Since the software was designed primarily for engineering students, visualisations were extended to engineering mathematics problems. An applet visualising the response of a linear second-order differential equation, set in the context of a mass/spring/damper system, was produced. This allowed dynamic investigation of the system response resulting from changes in the system's parameters, initial conditions and driving force.

A slight rewrite of this code converted it into an applet visualising the response of an electrical LRC series circuit. The system response of a general second-order system with a step input was visualised in an applet allowing users to move the system's poles in the s-plane resulting in real-time changes in the system response plot and the system's governing equation and solution. Mohr's circle is a graphical method of solving stress/strain problems in which students (laboriously) plot solutions using scale diagrams on graph paper. This static approach does not readily lend itself to *what if…?* investigations and so a Mohr's circle applet was an obvious choice for inclusion in the project.

An example of the user interface of one of the applets, featuring the solution of a first order linear differential equation in an LR series circuit engineering context, can be seen in Figure 1.

With *MathinSite*'s applets helping to deepen insight by giving users an *immediate*, dynamic feel for the effect of parameter changes and helping to facilitate *what if…?* investigations, keyboard data entry was deemed too slow and inappropriate. Consequently, *scrollbars* were implemented to effect dynamic parameter changes, with resultant *real time* changes in the graphics as the sliders are moved. The applet in Figure 1 shows five active scrollbars on the right hand side of the user interface. With some of the applets, it was judged more appropriate to achieve this by dragging points within the graphics area.

THEORY SHEETS AND TUTORIAL WORKSHEETS

MathinSite's starting point is the student who has already covered each topic in lectures. However, where it may help students to deepen their understanding by being reminded of the underlying theory, some

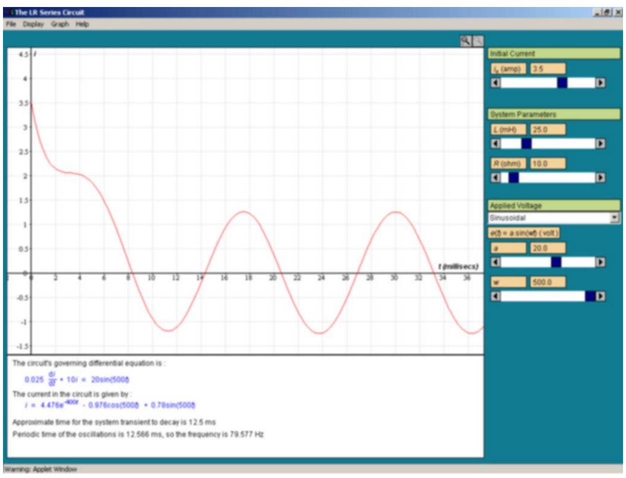


Figure 1: The LR series applet.

of applets have accompanying theory sheets. On the other hand, *all* applets are accompanied by a tutorial worksheet (or have worksheets in preparation). *MathinSite*'s theory and tutorial worksheets are stored on the Web site in portable document format (.pdf), thereby reducing document size and hence download times. These can be viewed under most operating systems using Adobe's freely available *Acrobat Reader*. More recent Web browsers also contain a .pdf viewer plug-in that allows documents to be viewed directly from inside the browser. Either way, users can freely download and print worksheets to use in parallel with the applets, so forming a permanent, takeaway record of their *discoveries*.

When writing the tutorial worksheets, a philosophy of *I know what I want you to see* was ruthlessly and unashamedly adopted, so totally prescribing the student experience. It was felt that total freedom in applet use would result in a lack of focus. A judicious choice of guiding questions ensures that the project's aims are achieved while still maintaining student motivation, interest and sense of discovery. Needless to say, this approach was not intended to preclude inquisitive students who want to play with the software. In fact, students are encouraged to use the software after having completed the worksheets in order to further their understanding.

Each applet's worksheets begin with some mathematical background and brief instructions on how to use the applet. Space is available on the worksheets to answer set questions by writing in the spaces provided on what is observed and sketching the graphical output from the screen. In encouraging reflective learning in this way, most students will:

- Make the required connections between changes made and the graphical outcome.
- Deepen their understanding.
- Obtain a greater *feel* for the mathematics involved.
- Remember the main points of the topic covered by the applet to a greater extent.

However, no matter how hard we try, there will always be those for whom this approach is ineffective!

IMPLEMENTATION CONSIDERATIONS AND PROBLEMS

Having *MathinSite*'s predecessor originally implemented as *Visual Basic* programs on non-networked PCs in the early 1990s resulted in having to replace the software regularly, since users were easily able to corrupt the set-up on each computer. Replacing the software repeatedly resulted in a lack of immediacy that proved unsuccessful in terms of gaining a good user base [9]. Although further *Visual Basic* programs were produced, the distribution of the software was held in abeyance.

This continued until it was realised that the Java programming language could provide an ideal, reliable and more secure way of packaging such programs – and for a much wider audience, since Java applets could be accessed and run from the Internet. The ability to allow access to the programs by any number of users around the world was an attractive incentive. Java, developed by Sun Microsystems during the 1990s, has become increasingly common on the Internet, since Java applets run within *freely available* Internet browsers. Not only that, but applets produced on a Windows PC, for example, will run through *Internet Explorer* or *Netscape* on a Mac or a Unix system (theoretically!). With this cross-platform compatibility and with individual installation problems now irrelevant, accessing applets through *MathinSite* is simple, apart perhaps from the time needed to initially download each applet on to the user's PC.

A lack of *absolute* cross-platform compatibility has proven to be a problem at times. This has ranged from *MathinSite*'s Web pages displaying differently on different platforms, to the more serious problem of inconsistencies across platforms in the presentation of the applets themselves. What is seen on one machine may not display the same way on another. Serious problems have occurred using plug-in code available from the Java *Abstract Window Toolkit* (AWT). For example, *MathinSite* makes much use of the *scrollbar* available directly to the Java programmer from the AWT. In early applets, it was noticed that, on some platforms, scrollbars would completely disappear when used! Even though Java has been consistently improved since its conception through Java versions 1.1, 1.2 and 1.3, for example, it has been important to ensure that *MathinSite*'s applets work on *all* computers with *any* operating system and *any* browser. Therefore all applets are written using Java 1.1, with each AWT component used tested on a variety of platforms. Even so, some AWT components have had to be replaced and rewritten in-house to try to ensure 100% compatibility – the scrollbar being a good example of such a *home grown* component.

The AWT is the standard Application Programmers' Interface (API) for providing a Graphical User Interface (GUI) for Java programs. Common Internet browsers, like *Internet Explorer* and *Netscape Navigator*, provide support for this API; no extra *plug-in* is required from Sun's Java site. However, later improvements in the Java language introduced

Swing, a new API, which provided many new GUI components - a very tempting prospect for mathematics applet developers! However, to use *Swing*-based applets, users are required to download and install a Java plug-in of about 5 Megabytes – a plug-in not available in all standard *Internet Explorer* or *Netscape* installations. It was realised that not all users:

- Would be aware of this requirement;
- Would know how to obtain the plug-in.

Also, downloading a 5-Megabyte file may be quick for users with broadband Internet access, but it would take too long for home-users of *MathinSite* with a 56k modem. So, although using *Swing* would have enhanced rapid application development, its non-use has ensured wider accessibility to *MathinSite*'s applets.

While the theory and tutorial sheets can be downloaded and stored on a remote user's PC and printed off for use with the applets, the applets themselves *have* to be run from *MathinSite*. This can cause problems. A class of 40 students sitting at 40 terminals in Tucson all trying to access *The Parabola* applet at the same time, for example, can cause the server to slow considerably. However, once downloaded onto the user's machine, running the applet is independent of the link to the *MathinSite* server. The speed with which the applet runs is then purely dependent on the local PC. One of the main requests to *MathinSite* is to release the applets for them to be stored and run locally without an Internet link. Mainly for copyright reasons and for logging *MathinSite*'s usage, this option has not been implemented.

Where does *MathinSite*'s software development go from here – and why ask? Since the mid-1990s, Java has been the only mainstream platform-independent language, but this is soon to be challenged by Microsoft's equivalent, .NET (*dot net*). .NET encompasses a wide range of services, but of particular interest is its new cross-platform, multi-language capabilities. Programmers' skills and knowledge of programming languages vary widely and, even with this diversity, more and more will want to publish their work on the Internet. To cater for this variety in skills, Microsoft, through .NET, is providing a common Integrated Development Environment (IDE), which is used to facilitate rapid application development using a special intermediate compiler for each language supported; this then compiles developers' work into multi-platform binary code. When a user requests an applet from the Internet, this binary code is downloaded, compiled by the just-in-time compiler on the local computer and then executed. The major difference with .NET is that the developer can use one or a combination of many different languages, such as *Visual Basic*, *Fortran*, *Visual C++* or Microsoft's new language *C#*, to write their applications, so not limiting Internet software developers to Java. Added to this is the fact that Microsoft's *Internet Explorer* version 6 does not automatically include direct support for Java and it begs the question, *in the meantime, should MathinSite developers be thinking in terms of .NET?* Watch this space.

MATHINSITE'S ADVANTAGES AND DISADVANTAGES

Many lecturers involved with the learning and teaching of engineering mathematics around the world may have already produced material that is presented in *MathinSite*, perhaps using computer algebra packages or graphical calculators. So is *MathinSite* just a case of *reinventing the wheel*?

In its ten-year development at Bournemouth University, Bournemouth, England, UK, user feedback has indicated a variety of reasons - some practical, some pedagogic - why *MathinSite*'s approach is popular. It is useful here, for comparison purposes, to put some of these reasons in a single list (but note that the list is not necessarily in ranked order of importance):

- *MathinSite*'s applets visualise mathematics: the dynamic, interactive visualisations can enhance and deepen the learning process and aid memory retention.
- The need to type in textual data is completely avoided. So speed, response and dynamic interaction are not compromised in any way.
- Each applet's immediate interactive feedback facilitates *what if …?* investigations.
- The accompanying worksheets guide students into *discovering* mathematics for themselves, thereby empowering them in their own learning process and heightening their own sense of achievement.
- Each applet is designed to be a self-contained single aspect of mathematics – bite-size chunks. Unless the applet is designed to do so, it is not possible to be sidetracked into other areas of mathematics.
- Applets cannot be changed by end-users. Each has a single task to do, points to make, and users cannot be distracted by altering the applets to perform other tasks.
- The software will not crash due to inappropriate data since scrollbars allow only parameter input from prescribed, valid ranges.

- The applets are easy to use. Each uses a standard Windows-type display (with which most students are familiar) and, further, all of the applets have a common user interface and require the same interaction. The only input is through scrollbars or by moving points around the graphics area. Using a computer algebra package or graphical calculator involves a learning curve just to access the software package or calculator even before the mathematics can be tackled (of course, for those already familiar with these, this would not be a problem.)
- The use of the applets does not have to be initiated by the lecturer. Their use does not even require supervision by a lecturer since the accompanying self-contained worksheets guide students through the applets' use.
- Since each applet is accompanied by a pre-written tutorial worksheet, lecturers can introduce them directly into their own tutorials or lectures without any further work. Obviously, the use of the worksheets provided is not mandatory and there will be those who will want to write their own.
- Although primarily a *learning* tool, *MathinSite* can be used, under certain conditions, as a *design* tool (eg designing LRC circuits to achieve a particular response).
- Lecturers can use the applets to design and/or check the answers of examination questions.
- With completed worksheets, users have a permanent record of their *MathinSite* sessions.
- The applets do not need proprietary software to run them apart, that is, from freely available *Java-enabled* Web browsers.
- The applets are immediately accessible anywhere on the planet at any time from any computer with an Internet connection.
- Their use is free.

Lecturers who have prepared similar material using proprietary mathematics packages may like to re-read the above list to determine whether their products satisfy *all* the above points!

MathinSite has its *disadvantages* too but, so far, these seem to be disadvantages seen only by the developers. These include:

- Despite object-oriented, reusable code and Java being described as a *rapid application development* tool, the production of applets can be slow.
- Production of *directed learning* material is time-consuming.
- The large amount of mathematical typesetting using *Word*'s *Equation Editor*, needed in some of the worksheets, is tedious.

PROJECT APPRAISAL AND FEEDBACK

The earlier, *Visual Basic* form of the applets had been used at Bournemouth University for several years with only partial success. With the set-up on each non-networked PC regularly corrupted, general use and, more particularly, out-of-hours student use, was severely curtailed. *MathinSite*'s delivery using the Internet has ensured that the *corrupted set-up* and a lack of general access are no longer a problem.

Despite uncorrupted, 24-hour access, a challenge that still exists is how to encourage students to use the software outside tutorials – there is always something more pressing to be done, especially around assignment hand-in dates! Some students *have* extended their tutorial work by subsequently using the software out-of-hours, others have needed only a gentle reminder that this facility exists, while a few, quite often those who need most help, have been reluctant, or cannot be bothered, to use the software at all.

One of the main purposes of the worksheets and their accompanying questions has been to encourage reflective learning, so it was disappointing to note during supervised tutorials that a few students rushed through the questions using them merely as *hurdles to be surmounted*.

A questionnaire circulated among students introduced to the applets during timetabled sessions indicated that most had gained a better insight into the particular aspects covered. They were positive in their encouragement of the production of further applets - naming particular topics they wanted to see in future developments.

Reviewing the applets has not been the sole domain of students. All academics at Bournemouth University were e-mailed about the new mathematics resource. Those who responded, from academic areas as diverse as business studies and nursing, were complimentary but, needless to say, more than willing to suggest topics that they would like to see included. Fulfilling such requirements is difficult, especially when the authors' own students were already the target market for *MathinSite*!

FUTURE DEVELOPMENTS

While it is hoped to support and extend the range of applets (be it with Java or .NET) after the current funding has ceased in 2003, there is no doubt that such an extension would be time-consuming – making further funding essential. However, the *what if…? + visualisation* approach adopted here could be applied to many academic areas and collaboration would be an alternative way forward. A further development could be to make *MathinSite* totally open-source, so allowing contributions to the site from other mathematicians around the world. At the time of writing, the future is bright but, as yet, unresolved.

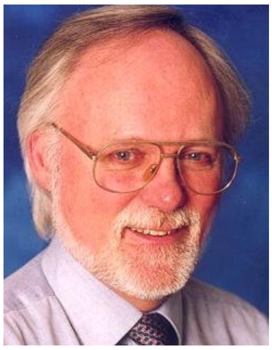
CONCLUSION

Feedback has indicated that *MathinSite*'s applets are popular, easy to use and *shedding light in dark corners*. They do not work for everyone; but then, books, printed notes, videos and lectures do not always work for everyone. The applets should, nonetheless, be seen as a valuable resource in promoting greater mathematical insight.

REFERENCES

1. Wulf, W.A., The urgency of engineering education reform. *Proc. ASEE Annual Conf.*, Montreal, Canada (2002), http://www.asee.org/conferences/annual2002/wulfplenary.cfm
2. Howson, A.G. (Chairman), Tackling the Mathematics Problem. London: London Mathematical Society, the Institute of Mathematics and its Applications and the Royal Statistical Society (1995).
3. James, D.G. (Chairman), Mathematics Matters in Engineering. Working Group Report, the Institute of Mathematics and its Applications (1995).
4. Sutherland, R. and Pozzi, S., The Changing Mathematical Background of Undergraduate Engineers - a Review of the Issues. London: the Engineering Council (1995).
5. Edwards, P., Some mathematical misconceptions on entry to higher education. *J. of Teaching Mathematics and its Applications*, 14, **1**, 23-27 (1995).
6. *Mathwise*, http://www.bham.ac.uk/mathwise/
7. *MathinSite*, http://mathinsite.bmth.ac.uk/
8. NTFS Awards, http://www.ilt.ac.uk/news/n20000712.html
9. Edwards, P., A visual approach to understanding mathematics. *CTI Maths&Stats Quarterly Newsletter*, 6, **1**, 2-7 (1995), http://www.bham.ac.uk/ctimath/reviews/aug95/visual.pdf

BIOGRAPHIES



Peter Edwards is Professor of Engineering Mathematics Education at Bournemouth University, UK. He has a first degree in mathematics from Liverpool University and two postgraduate degrees from Imperial College, London, one in aeronautics and one in robotics. Professor Edwards joined higher education after three years in the UK aircraft industry and has been teaching engineering mathematics for over 30 years. He has published widely on mathematics diagnostic testing and its follow-up support. In 2000, he was rewarded for his teaching excellence by being one of the first lecturers in the country to receive a UK National Teaching Fellowship.

During 2002-2003, Paul M. Edwards completed a BSc in Computer Studies at Bournemouth University, UK. During 2001-2002, Paul was employed on the *MathinSite* project during his undergraduate Industrial Placement year. Paul intends to make a career in the computer graphics industry.