ENGINEERING MATHEMATICS EDUCATION USING JAVA LEARNING ENVIRONMENTS

David I. Laurenson, Mike J. Jackson[1]

Abstract *¾ The Java language has been used in the construction of a number of learning environments to enable students to explore mathematical concepts without the barrier of mathematical manipulation. The applets are designed to be flexible for educators, and flexible in usage to encourage exploration and experimentation, yet are compact to minimize download time. The use of the developed environments in a practical education scenario is described, showing how the tools are used for class instruction, tutoring support, and individual learning. Subjective and qualitative evidence is presented showing that where the environments are used by students there is a discernible improvement in understanding of the concepts that the environments address. The results of this process are generalized to show the type of concept that is suitable for such treatment, and those for which improvements are not discernible.*

Index Terms ¾ customizable, experimental evidence, Java applets, mathematical concepts.

INTRODUCTION

Mathematics is at the heart of many Engineering problems, and a good understanding of mathematical principles allows Engineers to reduce a problem to a set of equations that may be solved. Thus a good grasp of mathematical representations of problems is required by students and practitioners of Engineering alike. However there is an increasing problem observed by many Universities, that the mathematical prowess of students is declining. This creates a problem for the Engineering educator who is charged with teaching how to apply mathematics to a problem when the basic mathematical manipulative and conceptual skills are not well understood.

This paper is based upon work at The University of Edinburgh to assist students in their understanding of common signal analysis tools such as the discrete Fourier transform (DFT) as well as tools for analyzing random processes. The premise for the work is that students should be able to explore the effects of altering parameters within the analysis tools, thereby increasing their understanding of the particular tool operation [1] without the hurdle of understanding the mathematical manipulative process first [2,3]. Many students find that the mathematics, far from increasing clarity of a problem, obscures the problem and shrouds it in notation that is unfamiliar, complex, and requires them to employ significant mental effort to understand. This detracts from them obtaining an understanding of the problem and how the mathematical process assists them in processing.

The learning environments that have been developed concentrate on the learning process through visualization. The hypothesis is that if the students are no longer involved in the process of developing a visualization of a mathematical process that they will be better able to concentrate on the problem itself. This should allow them to then apply the mathematical tool to other situations without necessarily understanding the mechanics in detail, but nevertheless understanding the effect of the processing. Through such application it is hoped that when the mathematical notation is used that links between the visualizations and the mathematics will become evident.

This paper will describe the set of learning environments that have been developed, with their key properties highlighted, and then proceed to investigate the evidence as to their effectiveness. Evaluation of the environments is carried out on a class of students being taught the material through subjective and qualitative evidence. Finally, some conclusions are drawn from the work with the effective key features of the learning environments identified.

JAVA BASED LEARNING ENVIRONMENTS

The aim of the project driving the development of the learning environments being described was to provide a set of tools that would operate on as many different platforms as possible, and be able to be delivered over the World Wide Web. To this end, the design of the learning environments was such that documentation could be included, in hypertext markup language (HTML) or portable document format (PDF), and experiment environments could be used. The Java language possesses many of the properties required – that of being supported by a wide variety of platforms, compact for downloading over the internet, and powerful enough to provide an experimental environment. Version 1.1 of the language was used in preference to later more powerful versions better suited to some of the graphical interfaces. The reason is that it is accepted that students will be using older machines with browsers unable to support later language versions without requiring additional software installation – an activity known to discourage the use of a learning facility [4].

[1] David I. Laurenson, Mike J. Jackson, The University of Edinburgh, Kings Buildings, Mayfield Road, Edinburgh, UK, EH9 3JL

Use of powerful learning environments can be bewildering for students, particularly if it takes a long time to learn what facilities are available to be used. For this reason instead of developing a single Java application that could be used for all learning tasks, a suite of applications designed to a common user interface have been developed that address specific issues. Within each application there is a further degree of customization possible whereby certain facilities can be selectively disabled, and particular sets of signals be chosen. Thus the same application can be presented in a variety of ways as desired by the educator.

The environments are designed to be used in three ways; as a lecturing tool; an interactive assistant for one-to-one tutoring; and as a self driven experimentation environment. The accompanying documentation is written with the last of these in mind. As the environments include a Java application which is displayed in one or more windows, the environment allows the tools to be scaled to fit the current screen resolution, or to smaller sizes as appropriate. Such flexibility simplifies the process of coordinating information between multiple windows of a computer screen so that students can draw on information from other sources.

The core element of all of the tools is the graph, both two dimensional and three dimensional. This is used to represent signals in the time domain, the frequency domain, the Laplace domain or the Wavelet domain as appropriate. Signals are shown as a continuous line, or as discrete bars for continuous and sampled data respectively. Multiple graphs are displayed in one application at one time. Where possible, whenever a change in one of the data sets is made corresponding changes are made to all other graphical displays. By using this technique the student can observe the effect of all actions in multiple domains simultaneously, and therefore develop a mental concept of interrelationships between domains.

The particular concepts selected for this treatment are:

- Convolution
- Correlation
- Ensemble averaging
- Fourier transforms
- Power spectral density
- Phasors
- Spectrograms
- Wavelets

Customisation of the learning environments is achieved through the use of an auxiliary text file. This specifies details such as the colours of the application components, the style of graphs to be displayed, including their scales where appropriate, the components of the application that are to be selectively enabled or disabled, as well as specifying the signals to be used within the tool. This flexibility has been used in practice to customise the environments for lecture presentation, both for style and for including lecture examples within the learning environment. Such flexibility ensures that the learning environments are able to be used across a variety of disciplines within Engineering.

EVALUATION OF THE LEARNING ENVIRONMENTS

In order to avoid the introduction of bias resulting from an artificial learning scenario, the learning environments were evaluated through their use as part of normal teaching. The applications were used during lectures to final year students, and also in one-to-one tutorial scenarios where the material is covered appropriately by one of the learning environments. The students were also able to access the learning environments by themselves, although there was no compulsory element in the course that forced them to do so. Thus there is a self selecting body of students who made use of the learning environments in their own time, and all students were exposed to them through lectures.

Informal feedback indicated that the Java applications were found to be useful by the students, with numerous comments encouraging further development of the learning environments. This positive feedback initiated a more formal evaluation process which is based upon subjective feedback from the students as to their own perception of an increase or decrease in understanding, as well as objective evidence collected through two open-ended class tests. A full description of the evaluation procedure is given in [3].

The two classes of evaluation participant reflected the two classes of user for which the tools were designed:

- The lecturer, who also undertook duties as a tutor
- Students taking the module

There were no set tasks for these participants and no control imposed on the environments in which the tools were used beyond the deployment of the tools and the resources to collect information from the participants. Students were informed that use of the tools was at their own discretion and their decision would not in any way be related to the course assessment. The participants undertook their standard teaching and learning activities using the tools as they felt appropriate.

Subjective Feedback

Subjective evidence was collected from the lecturer by interview, and from the students through on-line questionnaires that were completed after using the learning environments. It was made clear to students that the on-line information was anonymous, and that no information from this would be used in any way for assessment purposes.

A lecturer's observations can be a valuable information source when assessing the effects of educational tools. When asked if there had been any observed change in student behaviour since the deployment of the tools the lecturer stated that:

> *The students appear to understand the concepts more readily, even though some students require*

one to one tuition, with the tools, to finally grasp the concepts.

The lecturer then described how:

The questions that I am asked tend to be more involved than I have had in previous years, displaying a higher cognitive process than the basic operation of the equations. Even without revision, students are more able to understand what is being presented, and repeatedly tell me that they find the tools very useful in aiding understanding both during the lectures and when used by themselves.

These comments illustrate the effectiveness of the Java applications when used in the lecturing and tutorial scenarios as it is known that only a subset of the students used the learning environments for personal study. Even allowing for variations from year to year of the lecturing style, the dramatic change observed by the lecturer indicates that the learning process has progressed to a level deeper than in previous years.

Comments were elicited from the students through the use of a set of short on-line questionnaires. One questionnaire accompanied each class of learning environment and was completed after the student had experimented with the Java application.

Each questionnaire was simply structured asking the students whether they felt that their understanding had improved, remained the same, or become worse after using the learning environment. A section for written comments was made available for them to expand upon their response, or make any other comments they wished to.

Comments from the students are generally supportive of the learning environments stating that they found them useful:

[They] helped me to see for myself what was going on. Being able to play them over and over again was very helpful.

Students also made comments in which they explicitly mentioned that merely viewing the visualisations was useful but was no substitute for actual interaction:

It's good to look at the [tools] but it's better to try the things out and see how they're running.

An analysis of students' beliefs as to the effect of the tools on their understanding demonstrates that students overwhelmingly believed that using the tools has benefited them. Table I enumerates the responses from students stating their perception of how there understanding has been altered through use of one of the learning environments. Though there is a risk that some students believed their understanding had improved when in fact it had not – arising from the development of an understanding that is in fact flawed – it can be argued that for the majority of students their belief would be a reasonable reflection of their state of understanding.

TABLE I

STUDENT SUBJECTIVE ASSESSMENT OF UNDERSTANDING

Concept	Poorer	No change	Improved
Fourier Series (FS)	2	4	6
Fourier Transform (FT)	0	5	13
FS relationship to FT	0	3	9
Discrete Fourier Transform	0	3	9
Rotating Phasors	0	2	13
Correlation	0	3	7
Total	2	20	57

The different number of responses for each concept reflects that self study using the learning environments is optional, thus a number of students will have used some, but not all of the Java applications.

Clearly two students indicated that the use of a particular tool caused them to be more confused afterwards. One of these students explained that:

Things appeared on the complex plane with no explanation as to why. I know it was because I changed a parameter, but still.

It is interesting to note that the Java applications do not have any context sensitive help facility, relying instead upon good quality tutorial documentation. Obviously there is a need for a more interactive help facility with the aim of assisting students when they have some difficulty understanding a particular concept. An alternative solution would be to have academic staff available in a consultation role that students would be able to access when conceptual difficulties arise.

Objective Assessment

Subjective feedback is only valuable as a general indication of the learning environment effectiveness. Such feedback is subject to the students' perception of how well they understand a topic, and assumes that any increase in understanding is a correct increase in understanding, and not the reinforcement of a misconception. As highlighted above it is entirely possible for a student to become more confused using a tool, or worse that they think that they do have a correct understanding when in fact it is fundamentally flawed.

Short tests were used to determine the level of understanding of all students, with an indication on the test paper as to students who had used the learning environments for themselves, and those who had not done so. The short tests asked the students to write down as many points as they could on a given topic, or to specify the effect of an operation on a given signal. The tests were time limited to 10 minutes, and no indication given in advance of when the tests would take place, or on what topics, to avoid additional burdens on the students, and to test the underlying understanding at some point during the course. The test papers were not attributable back to the student, and the students understood that they were not being used for summative assessment.

Four specific topics were tested using these short tests:

- Windowing as used in the discrete Fourier transform (DFT)
- Zero padding as used in the discrete Fourier transform
- Continuous Fourier transforms
- The similarities and differences between correlation and convolution

Within each of these topics the student solutions were examined to identify whether a number of specific points were mentioned. Table II lists the results categorised into two groups – those who had used the learning environments for self study, and those who had not. The number of students completing the questionnaire is given as the first line in a section; the remaining entries list the percentage of students in that category who correctly identified the listed point.

TABLE II
RESULTS OF SHORT TESTS

Point	Description	Used	Not
	Windowing and the DFT	10	42
1	Window functions are applied in the time-domain	30%	21%
2	Window functions are used to reduce leakage	50%	19%
3	Window functions allow the discovery of frequency information that might otherwise be hidden	60%	15%
4	The transform of a signal is convolved with the transform of a window function, then sampled to give the DFT	30%	10%
	DFT and Zero-padding	10	42
5	Adding more samples to a sampled signal changes the window shape	0%	0%
6	Adding zero-padding interpolates values between frequencies allowing more detail of the DFT to be seen	60%	29%
7	Zero-padding involves adding zero value samples to the end of a signal	60%	81%
8	Zero-padding can allow circular convolution to work correctly	40%	43%
	Fourier Transforms	7	27
9	A time-shift causes a phase change in the Fourier transform	71%	59%
10	There is an inverse relationship between periodicity and sample spacing	71%	41%
11	providing a general sketch of the shape of a Fourier transform of a signal	43%	41%
12	providing the values of points where the signal crosses X-axis	14%	19%
	Convolution and Correlation	4	30
13	Convolution time-reverses one signal of two signals whereas correlation does not	100%	53%
14	One signal slides through another	100%	63%
15	Two signals are multiplied	75%	60%
16	The product of two signals are integrated	50%	56%

The table shows a clear distinction between concepts that are directly visualised by the tools and concepts that are more abstract in nature. Concepts that manifest perceivable phenomena include points 2, 3, 6, 9, 10, 13, 14 and 15. Students who had used the tools were more likely to answer these questions correctly, the difference in performance ranging from 12% to 47%. For question 15, though not so readily perceivable from the associated tool over 50% of students who used the tools answered this question correctly and performed better than non-tool users.

For more abstract concepts that are less readily visualised – questions 1, 4, 5, 8, 11, 12 and 16 – whilst the performance of students who had used the tools was in general better than those who did not, the results were much closer together, the differences ranging between 0% and 20%.

Questions 8, 12 and 16 elicited more correct responses from non users than users, the difference being within 6%, and therefore not statistically significant. The only notable instance of this was for the directly-perceivable phenomena of zero-padding involving the addition of zero-valued samples to the end of a signal (question 7) where 60% of students who had used the tools provided the correct answer compared to 81% of students who had not. In this final case it would appear that the tool, instead of aiding in understanding, has contributed to a confused understanding of the concept. In this particular case it is probably due in part to the layout of information on the screen, and also the process of animation. A user may not appreciate all of the changes that take place on-screen, a problem that may be addressed with improved supporting documentation.

DISCUSSION

There are many topics involving mathematics that engineering students have great difficulty understanding. Of most difficulty are topics where it is hard to relate the mathematics to an observable physical phenomenon or property. The use of mathematics in engineering can be taught in one of two ways – either as a formal subject where mathematics is an end in itself, and the student concentrates on proofs, or as application of mathematical tools to solve practical problems and analyse real systems. This latter form of mathematics, the one being addressed by the Java learning environments, proves to be the harder to teach as it involves the student not only being able to perform mathematical manipulation, but also the ability to relate the processes involved to the problem being solved.

Students using the learning environments have commented that:

> *A lot of the problem with learning Signal Analysis is that you cannot visualize what the lecturer is saying, and follow the math at the same time. Being able to see the signals means you can*

Clearly for this student the use of the Java applications in lectures has made the topic more readily accessible as the student is not involved in forming a personal visualisation the first time the topic is met. Results from the subjective and objective analysis do indeed reveal that the use of visualisations of mathematical processes does indeed improve the learning process, providing the student with the opportunity of gaining a deeper understanding of the topic.

Creation of a self-learning environment is a more complex task as it is very important to construct a support framework. The student must not feel that they have no recourse to a help resource if they find a topic confusing. For the learning environments developed the help resource is an HTML tutorial, with equations, which guides the students through a tutorial style learning task. For many students this was found to be adequate when access to the lecturer is possible. For some this was not adequate, thus the addition of a context sensitive help environment is required for the learning environments.

Despite the low numbers of students involved in the exercise, it is clear that there is not only subjective evidence that the use of the learning environments aids understanding, but there is also objective evidence that this is the case. Given that students approach problems in different ways, particularly when approaching conceptually difficult topics, offering a learning environment that performs the task of visualisation for the student will benefit some, may confuse others, and yet others will have their already good understanding reinforced.

Obviously the precise details of a given learning environment will affect the process of visualisation. In this particular instance the discrete Fourier transform tool does not represent zero-padding in an immediately understandable way, which impacts upon the learning process. It would appear that the animation style chosen has the effect of degrading understanding as opposed to improving understanding. However, an alternative explanation for this confusion can be given as the term also applies to the technique of convolution in the frequency domain. Here zero padding, the adding of zeros to a signal, is used to avoid circular convolution, not to show more detail of the windowed signal. Clearly such confusion could be addressed through context sensitive help, and by again examining the presentation of zero padding from a human computer interaction (HCI) perspective.

The Java learning environments have not attempted to assist in the learning of concepts that are not readily visualised. Despite this, experience has shown that such topics are more readily taught once the environments are used as the foundational elements, which may be visualised, are more readily grasped. The reason for this is that the more abstract concepts require a good working knowledge of the fundamentals, which is well provided for by the learning environment visualisations.

CONCLUSIONS

This paper has described a set of Java learning environments used to teach mathematical processes and concepts to final year undergraduate students in Engineering. The environments have been found to be helpful for lecture delivery, for tutorial support, and for student directed learning. In particular the environments are well suited to teaching concepts that can be visualized, and also assist in the process of learning more abstract concepts, less suited to the visualization process.

Both subjective and qualitative evidence has been used to show that the environments are successful in their aim of improving the learning process in the majority of cases. Such evidence encourages the further development of software specifically aimed at visualising mathematical concepts.

However, a need has been identified for improved on-line support. Provision of tutorial style documentation, although found helpful by many, was not sufficient in all cases to explain the observable phenomena to the students using the environments for self learning.

ACKNOWLEDGEMENT

The authors would like to acknowledge funding from the Scottish Higher Education Funding Council (SHEFC) who supported the evaluation exercise and much of the software development through the ScotCIT initiative. The materials from this exercise are available free of charge from the website http://www.webeng.org.

REFERENCES

[1] Bocker, H., Fischer, G., Nieper, H., "The Enhancement of Understanding Through Visual Representations", Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'86), April 1986, pp44-50

[2] Jackson, M., Laurenson, D., I., Mulgrew, B., "Supporting DSP Education Using Java", Proceedings of the First IEEE Signal Processing Education Workshop, October 2000

[3] Jackson, M., J., Laurenson, D., I., Mulgrew, B., "Developing and Evaluating Java-based Educational Tools", Proceedings of the First IEE International Symposium on Engineering Education, Vol. 2, March 2001, pp. 26/1-26/6

[4] Crutchfield, S., G., Rugh, W., J., "Interactive Learning for Signals, Systems and Control", IEEE Control Systems Magazine, Vol. 18, No 4, August 1998, p88-91