Alleviating the problem of Key Skills in Pure Mathematics modules using ICT

Dr Pamela Cowan

Graduate School of Education, Queen's University Belfast

P.Cowan@qub.ac.uk

This paper is based on a presentation from the conference ALT-C2001

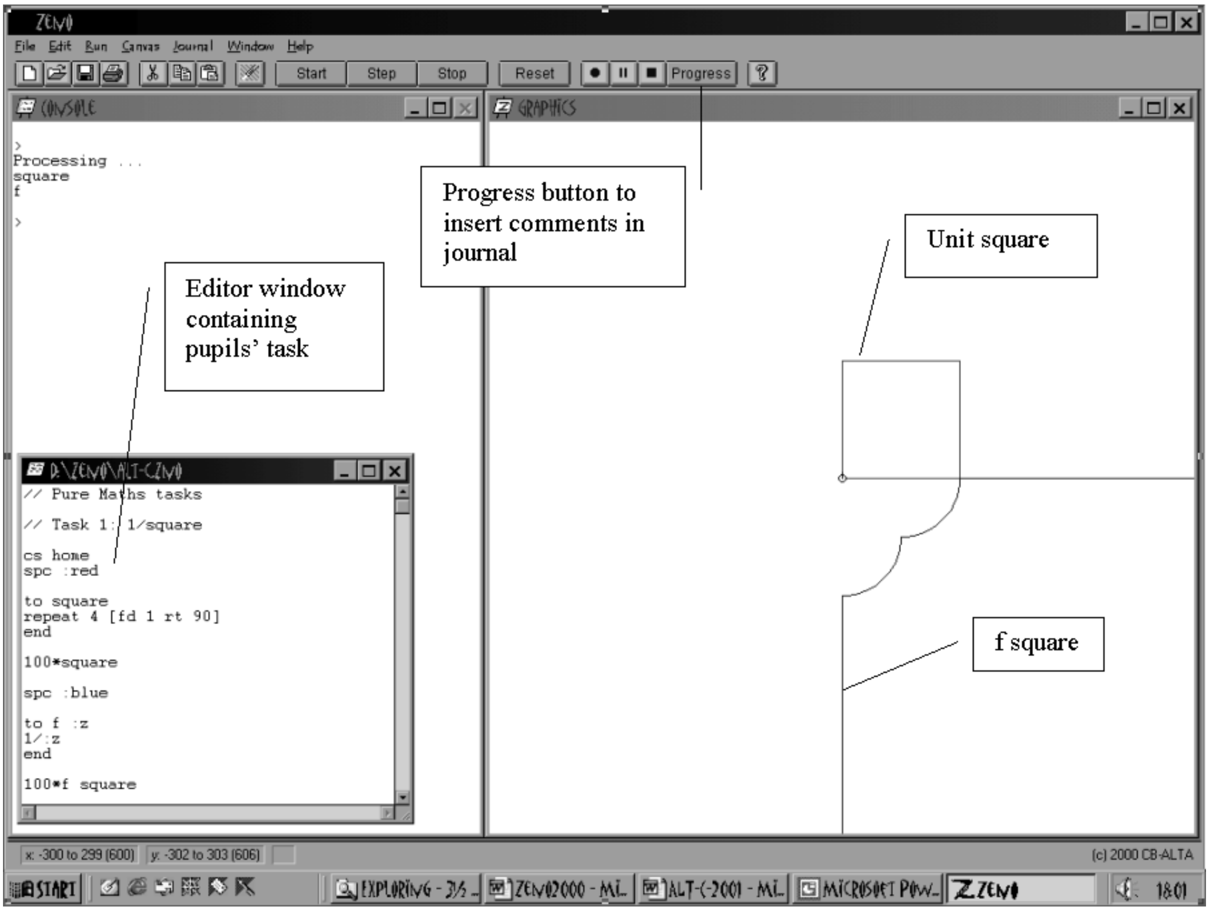
There has been much debate around the "value" of Key Skills and the ability to incorporate them into existing schemes of work at AS/A level. This article attempts to investigate the case of Pure Mathematics and to consider how the use of ICT could assist teachers in alleviating the current problems of incorporating Key Skills into their teaching schemes 'without distorting the integrity of individual subjects…' (Dearing, 1996, p 52)

This article proposes a computer-mediated solution to the problem surrounding the incorporation of problem-solving and information technology into the Advanced Level mathematics curriculum. It demonstrates that by using the functional programming language Zeno – which permits pupils to manipulate both the real and complex planes using a Logo-like functional programming language – pupils can solve mathematical problems of relevance to any Advanced Level mathematics syllabus, without compromising mathematical integrity. The journals generated automatically in the system contain rich evidence for inclusion in the Key Skills portfolio for both problem-solving and IT. However it is worth noting that the pupils' problem-solving skills (conjecturing, testing, generalising, controlling variables, working systematically, asking "what would happen if…?" questions and so on) have to be inferred from the content of the journal. In the current system the construct 'problem-solving' cannot be 'measured' although developments in this area are being investigated.

The Zeno software

Zeno is a functional programming language whose roots lie in Logo. The software was intended to offer pupils of secondary school age the opportunity to develop and extend their approach to problem-solving using a computer-mediated environment. It was designed through collaboration between the Graduate School of Education at Queen's University, Belfast and the Institute of Computer-based Learning at Queen's. While Logo restricts mathematical discovery to the real plane, Zeno spans both the real and the complex planes. The usual directional commands such as **forward** (**fd**), **backward** (**bk**), **left** (**lt**) and **right** (**rt**) followed by a numerical value are used to manipulate the turtle. These commands were chosen to ease the pupils' transition from Logo situations to Zeno tasks. Repeat loops and procedures are also paralleled in the Zeno software as is the use of variables as parameters in procedures. The main additional elements of Zeno are the increased number of available turtles, the new primitives for transformational geometry such as **reflect** (**rf**), rotate (**rr** to **rotateright** and **rl** to **rotateleft**), **translate** (**xl**) and **enlarge** (**enl**), the ability to represent and manipulate matrices, and the recognition of **i** in the vector format (0|1) where the real and imaginary parts of a number are split by the operator |. A more detailed overview of the Zeno language is offered in Morrison *et al.* (1999) for the interested reader.

The 'burden of awareness' is a measure of the degree to which a pupils' efforts to come to terms with the programming language obstruct rather than facilitate his or her mathematical enquiry. While this burden is great, the pupil cannot make progress without frequent reference to the programming manual; when the burden is light, the language is natural in that it closely resembles the symbolic language used in standard school mathematics texts. Users of the Zeno software have found the language easy to master even when their use of LOGO in junior forms was minimal.



```
>
Processing ...
square
f

>
```

```
// Pure Maths tasks

// Task 1: 1/square

cs home
spc :red

to square
repeat 4 [fd 1 rt 90]
end

100*square

spc :blue

to f :z
1/:z
end

100*f square
```

Progress button to insert comments in journal

Editor window containing pupils' task

Unit square

f square

The Zeno environment is completely open-ended. There are no pre-defined microworlds and students have access to the entire range of primitives in the system at all times. Depending on the topic being covered in class, the Zeno environment can be used to support teaching or to extend the content coverage through using a teacher-defined investigation or 'problem-solving'. The following example is designed to demonstrate Zeno's capacity to elicit evidence of inductive reasoning and problem-solving in an area of mathematics just outside the standard Advanced level domain.

Task 1:

The function $f(z) = z^{-1}$ is applied to a square in the Argand diagram.

The procedure

```
to square
   repeat 4[fd 1 rt 90]
end
```

draws a unit square. This unit square is magnified by a scale factor of 100 to make it visible in the Graphics window:

```
100*square
```

The mathematical function $f(z) = z^{-1}$ is represented in the procedure **f** :

```
to f :z
   1/:z
end
```

where z is the input variable.

In the next line, the function **f** is applied to the unit square (**f square**) and the image is magnified by a scale factor of 100 once again to ensure it is visible on-screen.

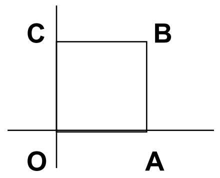
The pupils will notice that the two sides containing the origin are transformed to semi-infinite lines while the other two sides are transformed to quarter circles. To demonstrate their mathematical problem-solving skills, the pupils are then required to explain why this has occurred. Using their understanding of the mathematics involved, the pupils can break this problem into its four component parts and explain what is happening for each line in the square separately.

One possible explanation would be:

Each point z=x+iy is mapped to a point on the w-plane where w=u+iv.

Then u + iv = $\frac{1}{x+iy} = \frac{x-iy}{x^2+y^2}$

So u = $\frac{x}{x^2+y^2}$ and v = $\frac{-y}{x^2+y^2}$



OA: 0 ≤ x ≤ 1 and y = 0 therefore u = $\frac{1}{x}$ and v = 0

OC: 0 ≤ y ≤ 1 and x = 0 therefore u = 0 and v = $\frac{-1}{y}$

Consequently looking at the range of values for x and y respectively, OA maps to a line extending from (1,0) along the positive real axis and OC maps to a line extending from (0, -1) along the negative imaginary axis.

By looking at the remaining two lines, AB and BC, the two quarter circles must result from these two lines. What mathematical explanation can be offered in these cases?

AB: 0 ≤ y ≤ 1 and x = 1

therefore u = $\frac{1}{1+y^2}$ and v = $\frac{-y}{1+y^2}$

Eliminating y leaves $(u - 0.5)^2 + v^2 = 0.25$ which is the equation of a circle with centre (0.5, 0) and radius 0.5.

With A = (1, 0) and B = (1,1), these two points are mapped onto (1,0) and (0.5, -0.5)

BC: 0 ≤ x ≤ 1 and y = 1

therefore u = $\frac{x}{x^2+1}$ and v = $\frac{-1}{x^2+1}$

Eliminating x leaves $u^2 + (v + 0.5)^2 = 0.25$ which is the equation of a circle with centre (0, -0.5) and radius 0.5.

With B = (1,1) and C = (0, 1), these two points are mapped onto (0.5, -0.5) and (0, -1).

In the interests of generalisability theory, teachers would be advised to use the Zeno environment in a number of contexts to allow pupils to demonstrate their 'true' ability via a variety of problem-solving tasks. Previous research has shown high levels of task dependency in the scores of pupils' work (Cowan, 2000).

Mapping tasks to the Key Skills of ICT and Problem-solving

In terms of completing mathematical investigations, pupils tend to follow the four Polya stages of problem-solving namely, identifying the problem, devising a plan, carrying out the plan and looking back. The sequence of events overleaf describes the problem-solving process carried out by sixth form pupils when completing the above tasks. Each step has been mapped onto the Key Skill criteria for IT or problem-solving at Level 3 (CCEA, 2000).

Conclusions

The four stages in programming - identifying the variables, writing the code, running the program, and debugging - exactly mirror the four Polya (1945) stages: understanding the problem, devising a plan, carrying out the plan and looking back. A case can therefore be made that Zeno can play a unique role in transforming traditional mathematics curricula to post-Dearing curricula which combine academic learning with Key Skill development. Zeno enables pupils following Pure Mathematics modules to demonstrate attainment in both problem-solving and information technology. While Porkess' (1997) suggested activities are 'bolt-on', Zeno tasks adhere rigidly to Dearing's (1996) exhortation for curricular integrity.

This article demonstrates that Zeno facilitates the discovery of ideas that have an important place in mathematics. Clearly pupils could take alternative tasks closer to the content domain at Advanced Level such as exploring the connections between linear transforms, matrices and complex numbers or they could have explored the fundamental distinction between real and complex numbers. Nonetheless, it follows that problem-solving in Zeno represents no threat to curricular integrity.

Turning now to the issue of teacher manageability, it would appear that teachers currently charged by legislation with teaching LOGO to Key Stage 3 and 4 pupils, should experience few difficulties presenting Zeno to sixth-formers. Indeed, it could be argued that the overlap between Zeno and LOGO at Key Stages 3 and 4 is complete so Zeno can be viewed as a natural progression into Advanced level mathematics. Previous research has indicated that familiarity with new Zeno commands or primitives required a minimal amount of

Pupils' practical activities	Key Skills: IT criteria	Key Skills: Problem-solving criteria
Identify the variables in the problem eg. z – the square shape	IT3.1(i)	PS3.1(i) PS3.1(ii)
Plan: Use the Zeno environment to view the geometrical output.	IT3.1(ii) IT3.1(iii)	PS3.1(i)
Use Zeno for testing hypotheses.	IT3.1(iii)	PS3.1(ii) PS3.1(iii)
Look in textbooks for an explanation of complex numbers or the complex plane.	IT3.1(ii) IT3.1(iii)	PS3.1(ii) PS3.1(iii)
Search the internet for sites discussing the transformational effects of the complex plane.	IT3.1(ii) IT3.1(iii) IT3.2(i)	PS3.1(ii) PS3.1(iii)
Check the accuracy of the websites in terms of bias.	IT3.1(iii) IT3.2(i)	PS3.1(iii) PS3.2(ii)
Post questions to a bulletin board or discussion group looking for recommended reading.	IT3.1(ii) IT3.1(iii) IT3.2(i)	PS3.1(ii) PS3.1(iii)
Take regular screen prints for report/presentation.	IT3.1(iii)	PS3.2(ii) PS3.2(iii) PS3.2(i)
Use Progress button frequently to record hypotheses and tests in journal.	IT3.1(iii)	PS3.2(ii) PS3.2(iii) PS3.2(i)
Carrying out the plan: Use the Zeno environment to create a procedure to draw a unit square (automated routine).	IT3.2(i) IT3.2(iii)	PS3.2(i)
Try different scale factors for the square.	IT3.2(i) IT3.2(ii) IT3.2(iii)	PS3.1(i) PS3.2(i) PS3.2(ii)
Include printouts of the graphics window as evidence of each stage.	IT3.2(ii) IT3.3(i)	PS3.2(ii) PS3.2(i)
Record conjectures and comments on the outcomes in the journal using the Progress window.	IT3.2(ii) IT3.2(iii) IT3.3(i)	PS3.2(ii) PS3.2(iii) PS3.2(i) PS3.3(i)
Discuss findings with teachers/peers.	IT3.2(i) IT3.2(ii) IT3.2(iii)	PS3.2(ii) PS3.2(iii)
Use online discussion groups or bulletin boards to confirm findings and extend knowledge of the subject area.	IT3.2(i) IT3.2(ii) IT3.2(iii) IT3.3(i)	PS3.2(ii) PS3.2(iii)
Looking back: Reflect on the findings.	IT3.2(iii) IT3.3(i) IT3.3(iii)	PS3.3(iii) PS3.2(ii) PS3.2(iii)
Create generalisations as appropriate.	IT3.3(i) IT3.3(ii) IT3.3(iii)	PS3.3(I) PS3.3(ii) PS3.3(iii)
Generate mathematical justifications of the findings or create mathematical proofs to explain outcomes.	IT3.3(ii) IT3.3(iii)	PS3.3(I) PS3.3(ii) PS3.3(iii)
Produce a report/powerpoint presentation to disseminate findings to teachers/peers/wider audience (online) taking account of audience and level of mathematical expertise.	IT3.3(ii) IT3.3(iii)	PS3.3(I) PS3.3(ii) PS3.3(iii)
Include images and numerical justifications from the journal.	IT3.3(ii) IT3.3(iii)	PS3.3(I) PS3.3(ii) PS3.3(iii)
Include a mathematical proof of findings in report/presentation.	IT3.3(ii) IT3.3(iii)	PS3.3(I) PS3.3(ii) PS3.3(iii)

teaching time (Cowan, 2000) so facilitating the integration of the software into the normal classroom teaching scheme.

Pupils also found the Zeno environment easy to use and adapted quickly to the syntax (Cowan, 2000). They responded positively to its incorporation into the mathematics curriculum at A/AS level and found their experiences working in the environment enlightening and enjoyable overall.

Acknowledgements

Grateful thanks are extended to Dr Hugh Morrison from the Graduate School of Education and to the members of the Institute of Computer-Based Learning at Queen's University, Belfast, for their involvement in the creation of the Zeno software and its journalling facilities.

Copies of the ZENO software can be requested from Dr Pamela Cowan (p.cowan@qub.ac.uk) or Professor Fred McBride (f.mcbride@qub.ac.uk).

Bibliography

CCEA (2000). Key Skills specification and support materials. Belfast: Clarendon Dock.

Cowan, P and Morrison, H (1998). Evidence of a spiral curriculum using a mathematical problem-solving tool. *Interactive Learning Environments*, 6, 3, p. 205-224.

Cowan, P (2000) A computer-mediated solution to the current problem of integrating Key Skills into the UK Advanced level mathematics curriculum. Paper presented at AERA, New Orleans.

Dearing, R (1996). *Review of Qualifications for 16-19 Year Olds*. London: School Curriculum and Assessment Authority.

Morrison, H, Cowan, P, McBride, F and McBride, C (1999). Enhancing mathematical problem-solving through Zeno. *International Journal of Mathematical Education in Science and Technology*, 30, 5, p. 661-673.

Polya, G (1973). *How to Solve it*. Princeton: NJ: Princeton University Press.

Porkess, R (1997). *Key Skills with 'A' Levels*: *Mathematics*. Coventry: RSA Examinations Board.