Manipulator or Magician: Is there a Free Lunch?

Peter Galbraith
The University of Queensland
<p.galbraith@mailbox.uq.edu.au>

Mike Pemberton
The University of Queensland
<mrp@maths.uq.edu.au>

We report on research into outcomes that emerge when MAPLE software forms a central component of both the lecture presentation and laboratory activity in a first-year undergraduate mathematics course. We have noted an increase in both the number and type of questions asked by students compared with conventionally taught classes, and have identified factors associated with a symbolic manipulator environment that appear to link with task demand and student success. While enabling capable students to move faster and further, our evidence suggests that the use of such software does not compensate for, replace, nor render irrelevant, flaws in mathematical background.

Undergraduate mathematics courses in Australia have for some time been integrating symbolic algebra software into their teaching programs (Pemberton, 1997). This local reflection of an international trend, has received impetus from national reports such as Mathematical Sciences-Adding to Australia (NBEET, 1996). This report noted that the mathematical sciences are becoming increasingly laboratory based, with significant implications for how they will be taught. It recommended that mathematics departments re-design courses to make best use of this increased computer power which is heavily centred around the impact of symbolic manipulator packages, such as DERIVE, MATHEMATICA, and MAPLE.

Computer-Based Undergraduate Programs

The form of computer-based instruction varies widely. Olsen (1999) describes one of the most extensive examples of automated instruction. She describes how politicians visiting Virginia Tech's Mathematics Emporium, a 58 000 square-foot (1.5-acre) computer classroom:

> see a model of institutional productivity; a vision of the future in which machines handle many kinds of undergraduate teaching duties-and universities pay fewer professors to lecture...On weekdays from 9 am to midnight dozens of graduate-student and undergraduate helpers can be observed strolling along the hexagonal pods on which the emporium's computers sit. The helpers are trying to spot the students who are stuck on math problems and need help. (p. 31)

This program is openly driven by economic rationalism. At the other extreme Shneiderman, Borkowski, Alavi, and Norman (1998) describe a model, in which electronic classroom infrastructure is extensive and expensive, containing full computer and multi-media facilities as well as designer courseware. Courses are scheduled into the electronic classrooms on a competitive basis, and successful applicants must guarantee that the resources will be used as designed. It is required that full use be made of the interactive, collaborative, multi-media environment. Between such extremes lie a variety of models of instruction, whose users are concerned in varying degrees about factory production on the one hand, and student understanding on the other. Of those valuing the latter Alavi (1994) directly imported constructivist principles into computer-based learning by emphasising that learning is best accomplished by acquiring, generating, analysing, manipulating and structuring information. However Templer, Klug, and Gould, (1998) raised problems in this arena, that were perceived to arise as a direct result of a symbolic manipulator (MATHEMATICA) environment. They noted that having mastered the rudiments, the majority of students "began to hurtle through the work, hell bent on finishing everything in the shortest possible time". The following comment (or a close relative), was noted as occurring frequently: "I just don't understand what I'm learning here. I mean all I have to do is ask the machine to solve the problem and it's done. What have I learned?" Kent and Stevenson (1998) on a similar theme saw both positives and negatives, questioning first whether computer-controlled mathematical procedures can be learned effectively without appreciation of their place in mathematics. Their evidence and observation suggested that unless some kind of breakdown in the functionality of some concept or procedure (say integration) was provoked, the student would not focus on the essential aspects of that concept or procedure. However they noted that the demands for formal precision that a programming environment places on its user, serves both to expose any fragility in understanding, and to support the building and conjecturing required of the re (construction) of concepts by learners. Templer et.al., also saw the symbolic manipulator environment of MATHEMATICA helpful in this respect, to the extent that its language was sufficiently close to that of mathematics for the two to be treated in tandem. Similar claims could be made on behalf of MAPLE and DERIVE. We note similarities to observations from earlier work, where in noting gains in some performance measures, reference was also made to dangers of reifying a "black box syndrome" (Park, 1993).

The program that is our focus of interest, is a mainstream course located between the extremes referred to above. It represents a model that sits within present university structures and resources. Issues associated with its implementation connect with those raised in Kent and Stevenson (1998), and Templer, Klug, and Gould, (1998). Like the latter we are concerned with the links between computer-controlled processes and their mathematical underpinnings, noting the similarities and differences between the respective symbolism. With the former we share an interest in the range of questions raised by students as they work with the software, as well as in their performance. Many examples we have noted reinforce that the three-way interaction between students, technology, and mathematics, creates interactions that a focus on examination performance alone is unable to address.

The Study Context

The first-year undergraduate mathematics subject, forming the context of this study provides for a population of 650 students studying within Science and Engineering degree courses. The teaching comprises a lecture series complemented by weekly workshops, in which approximately 40 students are timetabled into a laboratory containing networked computers equipped with MAPLE software. The lecture room is fitted with computer display facilities so MAPLE processing is an integral and continuing part of the lecture presentation. To support their workshop activity students are provided with a teaching manual (Pemberton, 1997), continually updated to contain explanations of all MAPLE commands used in the course, together with many illustrative examples. During workshops two tutors and frequently the lecturer also, are available to assist the students working on tasks structured through the provision of weekly worksheets. The students can consult with the lecturer during limited additional office hours, and unscheduled additional access to the laboratory is available for approximately 5 hours per week. The course is also available on the Web a medium that is attracting increasing custom. Solutions to the weekly worksheets are provided subsequently. The formal assessment in the subject is constrained by departmental protocol to a pen and paper test at the end of semester (90%) supplemented with a MAPLE based assignment (10%). Consequently to succeed students must transfer their learning and expertise substantially from a software supported environment to written format. This means that they must be able to develop understanding through the medium with which they work, while simultaneously achieving independence from it - involving the ability to learn and maintain procedures that a MAPLE environment does not enforce. The educational implications of this characteristic need pursuing in their own right, but additionally attention is focused on the relationship between the mathematical demands of tasks, and their representation in a MAPLE learningscape. With this in mind the following questions emerge as requiring attention, noting that they remain central questions whatever the relationship between symbolic manipulators and assessment.

1. What is the range of student generated questions that emerge when learning of mathematical content interacts with a symbolic manipulator environment?
2. Can structural properties associated with the MAPLE environment be identified that link task demand and student success?

Data Sources

The data for addressing these questions come from two sources. Tutors assigned to the MAPLE workshops were provided with diaries in which they entered, on a weekly basis, examples indicative of the range of questions raised by students in the course of their workshop activity. The second source of data was a test given 7 weeks after the program started. This test was a voluntary exercise, and comprised a series of questions to be addressed with the assistance of MAPLE in its laboratory context. It was intended to provide formative feedback to the students on their performance, and ranged from simple school level manipulations to new material introduced in the tertiary program. Three sample questions are included in the appendix, together with their MAPLE solutions. As an additional incentive the test was directly relevant to preparing for the formal assessment at the end of semester, for the procedures required are ones that the students need to be proficient with, irrespective of software support. Additionally several questions contained an explanatory component, where the students were required to interpret the meaning of graphical output. Two sets of data were obtained from the tests, which were analysed and marked by two of the course tutors using criteria designed by the researchers. One of these involved the recording of correct and incorrect solutions, except that for this purpose the quality or indeed presence of a final interpretation of graphical output was not taken into account. This meant that the correct/incorrect dichotomy was on the basis of MAPLE operations only. The second set of data was obtained from an analysis of the errors that led to incomplete or incorrect answers. These are elaborated below. It was necessary to restrict the questions included in the analysis because poor results on later questions may be attributable to cursory attention imposed by time constraints. On the basis of a review of the 250 (approx.) scripts submitted, it appeared we could assume that the first 16 questions had been attempted seriously by the group as a whole. For technical reasons two of these were deemed unsuitable for inclusion, so that responses to 14 questions formed the final data set.

Results

Error Classification

A combined total of approximately 180 questions indicative of the range of concerns displayed by students when working in a MAPLE environment was assembled. These have been classified into categories in decreasing order of prevalence, as shown below in Table 1, together with the respective percentages. Clearly an element of informed judgment is present in selecting categories and assigning questions to them. The number of questions per category varies from a maximum of 30 (15.6%) to a minimum of 10 (5.2%).

Table 1
Student Question Types

Question Category	Percentage
1. How to get result or use particular operation (HGRO)	15.6 %
2. Identify problem caused by a typo (TYPO)	15. 6%
3. Resolve syntax error (SYNTAX)	15.1 %
4. General request for help (STUCK)	11.5 %
5. Interpret a technical term (ITECH)	8.9 %
6. Answer a procedural query (PROC)	8.3 %
7. Problem assigning variables (VAR)	7.3 %
8. Problem specifying function (SPEC)	6.8 %
9. Query result (QRES)	5.7 %
10. Query meaning of symbol (QSYM)	5.2 %

The error analysis from the test results generated a range of individual flaws (over 600 in total), which could be coarsely grouped into four main categories as shown in Table 2. Again these are judgment based with an element of subjectivity-they are essentially errors of commission. Errors of omission, as evidenced for example by failure to invoke appropriate commands, could not be so readily quantified. Ignoring the last category, of an omitted verbal explanation, which has no necessary connection with MAPLE facility, we may examine the others in terms of their role in this mathematical context. As far as operations with MAPLE are concerned we may identity two categories (at least) of facility. One of these is the requirement of accurate syntactical representation of common elementary operations, such as are represented in the first row in Table 2. The other is the more sophisticated and demanding selection and specification of functions to achieve identified mathematical ends. Clearly there is interaction between mathematical understanding and function specification, for if the former is flawed the wrong selection may be made or functions combined inappropriately. Alternatively if the mathematics is correct, the desired outcome can be defeated by mistakes in the technical detail of function specification. Considering jointly rows 2 and 3 in Table 2 covers this interdependence.

Table 2
Test Error Classification

Category Title	Percentage
1. Syntax related errors [*, ( ), ^ etc.]	31.1 %
2. Function choice and specification errors	17.3%
3. Errors in mathematics	17.3 %
4. Comment omitted or inadequate	34.3 %

Regression Analysis

We sought to relate performance to the influence of the two categories just discussed, which have been labelled SYNTAX and FUNCTION respectively.

SYNTAX: refers to the general MAPLE definitions necessary for the successful execution of commands. These include the correct use of brackets in general expressions, and common symbols representing a specific syntax different from that normally used in scripting mathematical statements (such as *, ^, Pi, g:=).

FUNCTION: refers to the selection and specification of particular functions appropriate to the task at hand. Specific internal syntax required in specifying a function is regarded as part of the FUNCTION component, including brackets when used for this purpose. Complexity is represented by a simple count of the individual components required in successful operation. We now illustrate how these definitions work, by applying them to the examples given in the appendix.

Q2. SYNTAX: Incidence of ^ [2] *plus* * [2]; total=4.
FUNCTION: General structural form of factor(argument); factor [1] *plus* ( ) [1] *plus* argument entry [1]; total=3.
Q8. SYNTAX: Incidence of ^ [1] *plus* *[2] *plus* ( ) [2] *plus* x1[1] *plus* := [1]; total=7.
FUNCTION: General structural form of plot(function, domain); plot [1] *plus* ( ) [1] *plus* , [1] *plus* function entry [1] *plus* domain entry [1] *plus* domain specification [1]; sub-total=6.
General structural form of fsolve(function, domain); sub-total [5] *plus* domain specification[1]; total =12.
Q14. SYNTAX: Incidence of* [2] *plus* ( ) [3]; *plus* y [1] *plus* := [1]; total=7.
FUNCTION: General structural form of plot(function, domain); sub-total [5] *plus* domain specification [1];
General structural form of int(y, integ interval); sub-total [5] *plus* (subtraction) [1] *plus* integration interval specifications [2]; total=14.

Similar pairs were assigned to each of the 14 questions in the sample. Our diagnostic approach involves scoring on a correct/incorrect basis, as we are not (in this analysis) concerned with apportioning partial credit as would be necessary if grading student performance. The success rate on the questions is given by the fraction of students (N~ 250) obtaining the correct answer. We can regard these as providing a measure of the probability of success of a student from this group on the respective questions. For the questions in the Appendix the respective values are 0.89, 0.26, and 0.14.A linear regression analysis was performed using these probabilities as measures of the dependent variable (success), and SYNTAX and FUNCTION as input variables (Tables 3-5).

Table 3
Regression

Regression statistics	
Multiple R	0.8710
R Square	0.7586
Adjusted R Square	0.7148
Standard Error	0.1419
Observations	14

Table 4
Analysis of Variance

	Df	SS	MS	F	Sig F
Regression	2	0.6957	0.3479	17.29	0.0004***
Residual	11	0.2213	0.0201		
Total	13	0.9171			

Table 5
Other Statistics

	Coefficients	Standard Error	t Stat	P-value
Intercept	1.0947	0.0961	11.383	2E-07
SYNTAX	-0.0482	0.0168	-2.874	0.015*
FUNCTION	-0.0396	0.0122	-3.246	0.008**

Discussion

In reviewing the outcomes of our study we can relate to the comment of Ramsden (1997) that "the impact upon educational practice of powerful software…has been less profound than optimists hoped or pessimists feared". Almost all reports contain statements tempering enthusiasm with caution, or disappointment with optimism. A continuing challenge is articulated by Olsen (1999) following a description of the most extensive budget driven, automated, attempt at mass produced learning that we have so far identified.

> Instructional software issues are unlikely to be resolved quickly... If we want the software to help at all… it's got to understand how students might misconceive what is presented to them--and to figure that out from the student's response. And right now, only people do that well. (p. 35)

With respect to our first question the patterns evident in Table 1 reinforce that when students interact with mathematics through technology questions are generated rapidly, and their scope is vastly increased. We can identify at least four types of inquiry from the responses: Those that are simply procedural (what to do next); those that are mathematical in the traditional sense; those that are software related (syntax and symbols); and those generated by the interaction of mathematics with software (function choice and specification, interpretation of output). The intensity and scope of student questioning has ballooned in comparison with that in traditional practice classes, with software the major contributor through properties of fast processing and scope for formatting and specification errors, together with student initiative in exploring. Secondly, in examining the analysis relevant to our second question, we observe that while achieving more rapid and efficient closure to algorithmic procedures the use of MAPLE has not reduced the need for the mathematical attributes of understanding and attention to detail. We note this in the significant impact of the variables SYNTAX and FUNCTION on success rate. SYNTAX errors penalise those who lack sufficient care in expressing their work symbolically, while the demands imposed by FUNCTION are proportional to the principles and sophistication of the associated mathematics. On the other hand, for those students who possess conceptual understanding and due regard for precision, the MAPLE environment has provided a means to progress rapidly and successfully at a greater rate than could otherwise be achieved. So our conclusion to this point is that there is no free lunch (indeed laboratory tutors are lucky to get lunch at all). The propensity of students to alter their approach to reduce the learning potential available to them (Templer, Klug, & Gould, 1998) is apparent. It is hoped that as student performance is mapped more carefully, and lessons learned from their responses to both mathematical tasks and in teaching situations, new insights for teaching-learning options will be identified. New properties that emerge from the mutual interaction of students, mathematics, and technology can support new approaches extending beyond the models that thus far appear to have motivated many of the proponents of automated learning. Goals of doing faster and more cheaply that which was done formerly with blackboard, chalk, and paper are limited indeed.

References

Alavi, M. (1994). Computer mediated collaborative learning: An empirical evaluation. *MIS Quarterly, 18(2), 159-173.*

Kent, P., & Stevenson, I. (1999, July). *"Calculus in context": A study of undergraduate chemistry students, perceptions of integration.* Paper presented at the 23rd annual conference of the International Group for the Psychology of Mathematics Education, Haifa, Israel.

Olsen, F. (1999). The promise and problems of a new way of teaching math. *The Chronicle of Higher Education, 46 (7), 31-35.*

NBEET, (1996). *Mathematical sciences-adding to Australia.* Canberra: National Bureau of Employment, Education, and Training.

Park, K. (1993). A comparative study of the traditional calculus and mathematics course. *Ph.D.* University of Illinois at Urbana-Champaign.

Pemberton, M.R. (1996). *Introduction to Maple (revised edition).* Brisbane: University of Queensland Mathematics Department.

Ramsden, P. (1997, June). *Mathematica in education: Old wine in new bottles or a whole new vineyard?* Paper presented at the Second International Mathematica Symposium, Rovamiemi: Finland.

Shneiderman, B., Borkowski, E., Alavi, M., & Norman, K. (1998). Emergent patterns of teaching/learning in electronic classrooms. *Educational Technology, Research and Development, 46*(4), 23-42.

Templer, R., Klug, D., & Gould, I. (1998). Mathematics Laboratories for Science Undergraduates. In C. Hoyles., C. Morgan., & G. Woodhouse (Eds.), *Rethinking the mathematics curriculum* (pp. 140-154). London: Falmer Press.

Appendix: Sample Questions

(Questions in italics: MAPLE commands in bold: MAPLE output in ordinary type)

Q2. Factorize $x^3 - 6x^2 + 11x - 6$

Maple Solution

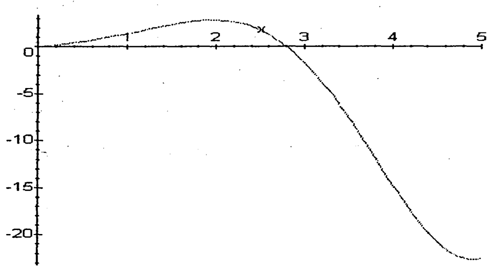
➢ **factor(x^3-6*x^2+11*x-6);**

Q8. Find where the graph of $x^2 \sin x + x \cos x$ *for* $0 \le x \le 5$ *is*:
(*a*) *above the* x – *axis* (*b*) *below the* x – *axis* (*c*) *cuts the* x – *axis.*

(x - 1) (x - 2) (x - 3)

Maple Solution

> **plot(x^2*sin(x)+x*cos(x),x=0..5);**



> **x1:=fsolve(x^2*sin(x)+x*cos(x),x=2..3);**

x1 :=2.798386046

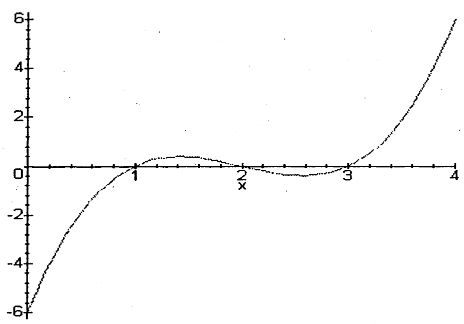
****

Q14. Plot the graph of f(x) = (x-1)(x-2)(x-3) and use this to find the physical area under the graph from x= 1 to x= 3.

Maple Solution

> **y:=(x-1)*(x-2)*(x-3);**

➢ **plot(y,x=0..4);**



➢ **int(y,x=1..2)-int(y,x=2..3);**

➢ 1/2