Using Maple to Learn *Mathematica*

Shirley Pomeranz

Department of Mathematical and Computer Sciences
The University of Tulsa

1. Introduction

This paper describes the use of the Computer Algebra System (CAS), Maple (Waterloo Maple, Inc.), to assist in the learning of CAS, *Mathematica* (Wolfram Research, Inc.) in a numerical methods course. What is novel here is that Maple was not used in a standard way. Instead, it was used in the following context. The students had (almost) all used *Mathematica* in previous courses, although the degree of skill in its use varied considerably. Numerical methods were taught as part of this course. Students had done previous numerical methods assignments in which a relatively simple problem was given and solved (or a formula derived) with *Mathematica.* Their assignments had been to use the given problem and *Mathematica* code as a template for using *Mathematica* to solve a related problem, which was more complicated.

For the final *Mathematica* assignment, the students were given a numerical methods problem that was coded and solved with Maple (see Figures I, II, and III). The students had no previous familiarity with Maple. They were instructed to use the given Maple code and solution as a template—to whatever extent they found helpful—in order to code and solve the same problem with *Mathematica*. They were also instructed to write brief reports describing their experiences with this use of Maple.

My purpose in giving this assignment was to see if students found the Maple template helpful. If they did find it helpful, then in what ways did it assist them? If they did not find it helpful, then what was problematic or confusing?

2. Background and motivation

During the previous year, I had been working on a *Mathematica* project, transcribing given Maple code into corresponding *Mathematica* code. I was part of a faculty team at The University of Tulsa (TU) that was working with the text, **Advanced Engineering Mathematics**, Robert J. Lopez[1]. Maple solutions to the computer problems in this text had already been written, and hard copies were provided to us. We did not have Maple running at TU, nor were any of us familiar with Maple. We used our knowledge of the mathematics, knowledge of *Mathematica*, and the fact that much of the basic structure and some of the commands in Maple were similar enough to those in *Mathematica* so as to be helpful. In this sense, the hard copies of the Maple solutions served as a combination template/answer key.

During my work on this project, I was learning more *Mathematica* programming. For example, we had a Maple worksheet that constructed, step-by-step, a D'Alembert's solution to the simplest form of the one-dimensional wave equation with zero Dirichlet boundary conditions. The initial position was identically zero, and the initial velocity was a piecewise-defined function. If one were to follow the Maple template exactly, the Maple command for constructing the piecewise-defined function, **piecewise**, would probably be replaced by the *Mathematica* command, **Which**. However, this would not work because we also needed to evaluate the indefinite integral of this piecewise-defined function, and *Mathematica* does not evaluate an indefinite integral of a function defined with a **Which** command. Thus, a general course of action has been prescribed by the Maple template, but the user now needs to experiment with several possible *Mathematica* commands that will not only conveniently construct a piecewise-defined function, but will also be accommodating to subsequent operations, such as indefinite integration. In this case, the user only needs to go to the *Mathematica* help browser and type in "piecewise". Information about the **UnitStep** function (Heaviside function) appears. This function can be used in the subsequent operations, including the indefinite integration, and it evaluates properly.

Continuing with this wave equation problem, a Maple procedure is given for computing a $2L$-periodic odd extension onto $(-\infty,\infty)$, of a given initial velocity function defined on $[0, L]$. This quantity is to be used in D'Alembert's solution. The commands in the procedure appear quite cryptic to one unfamiliar with Maple. However, by studying how the sequential Maple commands build on each other, and knowing that the intended outcomes are the construction of a $2L$-periodic odd extension and D'Alembert's solution, one obtains hints as to how to construct an analogous *Mathematica* module. (Here, first integrate, and then periodically extend.) Further, observing how complicated the Maple procedure is, leads the user to consider if there might be other easier ways to implement this construction in *Mathematica* (and Maple). (There is a tradeoff: some constructions are easier in Maple, and some are easier in *Mathematica*). There is an easier construction, another formula, which is equivalent to D'Alembert's formula here. This is the construction of the solution to the wave equation on $[0, L]$ that is formed by summing the infinite Fourier sine series that represents the $2L$-periodic odd extension of the initial velocity onto $(-\infty,\infty)$, and then forming the appropriate expression involving the integral of this extension. (Here, first periodically extend, and then integrate.) *Mathematica* easily evaluates this.

As another example, consider using separation of variables to solve Laplace's equation in a disk, i.e., in polar coordinates. Maple computed an approximate solution in polar coordinates and, with a little tweaking, plotted contour lines (level curves) and flow lines. This encouraged me to try to tweak *Mathematica* in some similar fashion. After observing that the Maple commands embedded a **changecoords** command, I got *Mathematica* to transform coordinates (using a **TrigExpand** command and replacement rules) and perform the desired contour plot using

Cartesian coordinates. In order to obtain the plot of the flow lines, the Maple commands "suggested" taking the *Mathematica* solution, which was also expressed as a function of polar coordinates, using the standard transformation equations from polar to Cartesian coordinates, and then feeding this composition into *Mathematica's* **ParametricPlot**. This worked fine.

I would be working on a repetitive batch of problems when I noticed a pattern in the results. I then went back to the algorithm that I was coding to see precisely how the results that I had observed depended on values of specific parameters. I admit that in some cases I was bored with what I was doing. I was not specifically trying to understand some aspect of the algorithm. I was working in a merely perfunctory manner trying to get things done (perhaps here I can identify with some students). Yet, the *Mathematica*/Maple work piqued my interest in "cause and effect". I learned something "in spite of myself". I'd experienced several "WOW" moments, and I was effortlessly learning more *Mathematica* programming **and** more numerical methods. For me, the combination of programming, graphics, and repetition, i.e., solving many similar problems, firmly anchored the knowledge in a way that hand calculations, etc., could not. This code translation process led me to insights about *Mathematica* programming and numerical methods. My own insights were the beginning of my idea to assign a similar project to my students. I became curious to know if my students might experience the same phenomena.

What would my students experience? They might not have the aspect of so much repetition, but they could be confronted with code and solutions in an unfamiliar programming language. How would they react?

3. Sample Maple-to-*Mathematica* assignment

The course, Introduction to Numerical Methods (MA4503/6603), taught in the Department of Mathematical and Computer Sciences at TU, is a standard introductory numerical methods course. Juniors, seniors, and a few graduate students take the course, and these students are from the various engineering departments and the Department of Mathematical and Computer Sciences. We used the text, **Numerical Analysis**, 7th edition, by Burden and Faires[2]. Every other week the class met in the computer lab and had a hands-on session with *Mathematica*. The following is a description of the sample Maple-to-*Mathematica* assignment.

I selected an assignment in which a goal was to show that a selection of centered finite-difference formulas for approximating various order derivatives of a function could be derived by differentiating appropriate Lagrange interpolating polynomials. This was based on material that we had already studied in class, and was assigned at the end of the semester—students had already spent most of the semester working with *Mathematica*. Students could work alone or in teams of two. See Figure I for the problem statement, Figure II for the given Maple solution[1], and Figure III for one possible corresponding *Mathematica* solution.

4. Student comments

As has been described, students were instructed to use the given Maple code and solution as a template in order to code and solve the same problem with *Mathematica*. They were also instructed to submit brief reports describing their experiences with this use of Maple. As might be expected, there was a range in the quality of the work that was submitted. Some of the assignments were virtually as efficient as the sample included in Figure III, while others contained almost nothing useful. The following is a selection of quotes from the students' reports.

- "It was relatively simple to understand the algorithm in the Maple code; however, it was a bit troublesome to convert the code line by line. We had to reevaluate our thinking process and convert the code block by block. The conversion of an idea was much easier to implement than converting a line of code. Once we understood the methodology behind a chunk of code, it was relatively straightforward to express that idea in *Mathematica's* syntax…. The reason why this process was straightforward was due to our knowledge of *Mathematica* on an intermediate level."
- "The Maple code was not very helpful as a template. Basically, the Maple code gave a general idea of what each objective required. Some commands in Maple have *Mathematica* equivalence, however, the structure of the code is very different…. Personally, we believe that if we had a better mastery of programming (i.e., *Mathematica*), the Maple code would serve a much greater purpose."
- "The help browser in *Mathematica* made it easy to find equivalent commands, once I had an idea what a specific Maple command did. For example, since I knew that **interp** was used to create an interpolating polynomial (in Maple), the help browser pointed me to the *Mathematica* command, **InterpolatingPolynomial**."
- "I mostly tried to perform the same calculations and emulate the lists with the *Mathematica* **Table** and **TableForm** commands."
- "Specifically, the code assisted in defining the flow and structure of the solution. It also assisted as an example of how the solution could be accomplished…. indicated that the interpolating function was to be the software's built-in interpolating module… this facilitated… the creating of a table of… data nodes and the interpolation of those nodes… Due to inherent differences in the Maple and *Mathematica* software, the use of the Maple code was sometimes confusing, especially when it was not necessary to imitate Maple statements in *Mathematica*, such as the use of the **unapply**, **evalf**, and **sort** functions."
- "The Maple loops shown in the handout helped in writing the **For** loops in *Mathematica*. In our opinion, the Maple helped more with the syntax of *Mathematica* than the actual implementation of the *Mathematica* itself. We just had to figure out how to convert between the two programs. However, since the template was not available in *Mathematica*, the problem could not just be 'cut and pasted' like some of the earlier problems this semester."
- "For part a), we found Maple to be very useful. It gave us several suggestions… For instance, the computation of the derivatives was much easier having the Maple code…. We could not determine what the **floor** statement was equivalent to in *Mathematica*."
- "Maple… gives excellent clues about the direction the computation should take, but does not provide significant information about entering the actual *Mathematica* code."
- "Trust me, I know the Help Browser VERY WELL after all this!"

One issue that may have been confusing for the students is the fact that they were used to receiving *Mathematica* templates from which they *could* copy the syntax of individual commands. Therefore, they may have incorrectly expected to be able to do so with this assignment also. Specifically, students were trying to apply the syntax for Maple's **interp** command to *Mathematica's* **InterpolatingPolynomial** command. These two commands have different syntax (with respect to the form in which arguments are supplied to the command). Students could not get the *Mathematica* version of the command to work when they incorrectly supplied the arguments in the form shown in the Maple template. I was disappointed that many students did not directly go to the *Mathematica* help browser, which we had been using all semester, and which would clearly give examples of command usage showing explicitly how the arguments were to be supplied. Instead, several teams of students showed up at my office and asked me why their code was not working. But perhaps they had learned the lesson of efficiently using the resources available and were thinking of me as just another type of help browser. Similarly, the students who stated in the above quote that they "could not determine what the **floor** statement was equivalent to in *Mathematica*", had not tried the *Mathematica* help browser; the equivalent *Mathematica* command is **Floor**.

5. Summary

Has anything new been presented in this paper? How is the process of giving students a programming template in a different (but structurally similar) CAS any different from providing a template in the same CAS?

Sometimes, if we see a problem solved one way, it becomes more difficult to forge a new problem-solving approach. There can be a middle ground between giving no clues at all versus giving clues that hinder original thought. One such middle ground has been presented in this paper. Namely, a template solution is given, but only in a general sense. The student uses the template in a general sense, and room is allowed for creativity. A general direction is provided, but with more latitude to "think outside of the box".

Another feature is that exposure to new and unfamiliar computer software tools will help prepare the student for what happens in the workplace. Engineers are exposed to new software, software for which they have not been specifically trained. Experience in adapting knowledge from familiar software to new related software could help students feel more comfortable when this situation arises in the workplace.

6. Acknowledgements

The author wishes to acknowledge and thank the students of MA4503/6603, fall 2001, for their hard work, collective sense of humor, and individual permissions to quote from their comments. Thanks also to Robert J. Lopez for his valuable text[1].

Figure I

Problem Statement

Math4503/6603 Mathematica Assignment:
Numerical Differentiation
Some Basic Formulas

Problem taken from Advanced Engineering Mathematics
Robert J. Lopez, Addison-Wesley 2001
Section 42.1, page 1065, problem 19

Assignment, part 1: Read the following problem.
The purpose of the following problem is to emphasize that each of the four central-difference formulas given below (*) is equivalent to (that is, can be derived from) the differentiation of an appropriate interpolating polynomial.

(*) Formulas:

$$f'(c) \approx \frac{1}{2h}\left(-f(c-h)+f(c+h)\right) \qquad \text{Eq. (a)}$$

$$f''(c) \approx \frac{1}{h^2}\left(f(c-h)-2f(c)+f(c+h)\right) \qquad \text{Eq. (b)}$$

$$f'''(c) \approx \frac{1}{2h^3}\left(-f(c-2h)+2f(c-h)-2f(c+h)+f(c+2h)\right) \qquad \text{Eq. (c)}$$

$$f^{(4)}(c) \approx \frac{1}{h^4}\left(f(c-2h)-4f(c-h)+6f(c)-4f(c+h)+f(c+2h)\right) \qquad \text{Eq. (d)}$$

For the nodes $x_k = c + k * h$, $k = 0, \pm 1, \pm 2, \ldots$, as appropriate, with $h = 1$, $c = -2$, and $f(x) = x e^x$, perform the following operations:

a) (i) Compute the exact first, second, third, and fourth derivatives of f evaluated at $x = c$.
(ii) Apply each of the formulas, Eq. (a) - Eq. (d), to approximate the relevant derivative at $x = c$.

b) Obtain $g_a(x), \Lambda, g_d(x)$, the polynomials interpolating points $(x_k, f(x_k))$ used for each differentiation formula in part a).

c) For each of $g_a(x), \Lambda, g_d(x)$, evaluate its appropriate derivative at $x = c$, and compare it to the corresponding result in part a). Show that you get the same results here, in part c), as in part a) (ii).

You are given the Maple code that solves this problem.

(Part 2 is worth 10 points)
<u>Assignment, part 2:</u>
Using the Maple code as a template or guide, you are to write *Mathematica* code that solves the problem.

(Part 3, if done thoughtfully, is worth 10 points)
<u>Assignment, part 3:</u>
Carefully write a short report discussing how the Maple code used as a template helped and/or didn't help you to write the *Mathematica* code. Describe any specific situations of interest, any related insights, or any relevant ideas that you have.

Figure II

Maple (Release 4) Solution
(From <u>Advanced Engineering Mathematics</u> [1])

Exercise 19

a

The function $f(x)$:

$f := x \rightarrow x^* \exp(x)$:
$'f'(x) = f(x)$;

$$f(x) = x^*e^{x}$$

Define:
$c := -2$;
$h := 1$;

$$c := -2$$

$$h := 1$$

The exact values of the derivatives $f^{(k)}(c), \quad k = 1,...,4$:

```
for k from 1 to 4 do
evalf ((D@@k) (f) (c));
od;
```

$$-.1353352832$$

$$0$$

$$.1353352832$$

$$.2706705664$$

Approximations of these derivatives via formulas (42.3a) thru (42.3d):

```
F[1] := evalf( 1/(2*h) * (–f(c-h) + f(c+h) ) );

F[2] := evalf( 1/(h^2)* ( f(c-h) – 2*f(c) + f(c+h) ) );

F[3] := evalf( 1/(2*h^3) * (– f(c-2*h) + 2*f(c-h) – 2*f(c+h) + f(c+2*h) ) );

F[4] := evalf( 1/(h^4) * ( f(c-2*h) – 4*f(c-h) + 6*f(c) – 4*f(c+h) + f(c+2*h) ) );
```

$F_1 := -.1092591180$
$F_2 := .0241004865$
$F_3 := .2551495139$
$F_4 := .371676632$

¦ b

The polynomials $g_\alpha(x), \quad \alpha = a, b, c, d,$ that interpolate the points used by formulas

(42.3a) thru 42.3d):

```
for n from 1 to 4 do
X := [seq(c+k*h, k= - floor(n/2 + ½) .. floor(n/2 + ½) ) ];
Y := map(f, X);
g[n] := unapply( evalf( sort( interp(X, Y, x) ) ),x );
print(`g` [n] (x) = g[n](x) ) ;
od;
```

$$g_1(x) = .0120502432\ x^2 - \ .0610581447\ x \ - \ .4409878299$$
$$g_2(x) = .0120502432\ x^2 - \ .0610581447\ x \ - \ .4409878299$$
$$g_3(x) = .01548652632\ x^4 + .1664171294\ x^3 + \ .623389862\ x^2 \ + .8403387004\ x$$
$$g_4(x) = .01548652632\ x^4 + .1664171294\ x^3 + \ .623389862\ x^2 \ + .8403387004\ x$$

¦ c

The following table displays the approximate values of the derivatives computed in part (a) by formulas (42.3a) – (42.3c), and the values obtained by differentiating the interpolating polynomials found in part (b).

```
S0 :=  [1, 2, 3, 4]:
S1 :=  [seq(F[n], n=1..4)]:
S2 :=  [seq((D@@n) (g[n])(c), n=1..4)]:
```

```
unassign('n');
H :=  [n, f ^ ` ` (n) * ` ` (c), g[n] ^ ` ` (n) * ` ` c]:
stackmatrix( H, augment( S.(0..2) ) );
```

n	$f^{(n)}(-2)$	$g_n^{(n)}(-2)$
1	−.1092591180	−.1092591175
2	.0241004865	.0241004864
3	.2551495139	.2551495132
4	.371676632	.3716766316

Figure III

Mathematica Solution

42.1 Basic Formulas

Initializations

The following code is designed to run in *Mathematica* 4.0.

Clear["Subscript" , "Global `* "];
Off[General:: "spell"];

Exercise 19

Part a)

The function **f(x):**

f[x_] = x * e^x;

Define:

c = −2;

h = 1;

The exact values of the derivatives $f^{(k)}(c), \quad k = 1, \Lambda, 4$:

Table[D[f[x],{x,k}] /. x → c, {k,1, 4}] // N // TableForm

-0.135335
0.
0.135335
0.270671

Approximations of these derivatives via formulas (42.3a) through (42.3d):

$$\mathbf{F[1]=\frac{f[c+h]-f[c-h]}{2h}\ //\ N}$$
$$\mathbf{F[2]=\frac{f[c-h]-2f[c]+f[c+h]}{h^2}\ //\ N}$$
$$\mathbf{F[3]=\frac{-f[c-2h]+2f[c-h]-2f[c+h]+f[c+2h]}{2h^3}\ //\ N}$$
$$\mathbf{F[4]=\frac{f[c-2h]-4f[c-h]+6f[c]-4f[c+h]+f[c+2h]}{2h^4}\ //\ N}$$

-0.109259
0.0241005
0.25515
0.371677

¦ **Part b)**

The polynomials $g_\alpha(x)$, $\alpha = a, b, c, d$, that interpolate the points used by formulas (42.3a) through (42.3d):

Do[

$$\mathbf{X = Table[\ c + k * h, \{ k,\ -Floor[\frac{n}{2}+\frac{1}{2}], Floor[\frac{n}{2}+\frac{1}{2}]\}\ ];}$$

Y = f[X] // N;

g[n][x_] = InterpolatingPolynomial[Transpose[{X, Y}], x];

Print["g [" , n , "][x] = " , g[n][x] // Apart // Chop],

{n, 1, 4}]

$g[1][x] = -0.440988 - 0.0610581\, x + 0.0120502\, x^2$

$g[2][x] = -0.440988 - 0.0610581\, x + 0.0120502\, x^2$

$g[3][x] = 0.840339\, x + 0.62339\, x^2 + 0.166417\, x^3 + 0.0154865\, x^4$

$g[4][x] = 0.840339\, x + 0.62339\, x^2 + 0.166417\, x^3 + 0.0154865\, x^4$

¦ **Part c)**

The following table displays the approximate values of the derivatives computed in part (a) by formulas (42.3a) - (42.3d) and the values obtained by differentiating the interpolating polynomials found in part (b).

```
S0 = Range[4];
S1 = Table[ NumberForm[ F[n], 10] , {n,4} ];
S2 = Table[ NumberForm[ D[ g[n][x] , {x, n} ] , 10 ]  /. x → c,  { n, 1, 4 } ];
TableForm[Transpose[ {S0, S1, S2} ],
TableHeadings- > { None, { "n" , "f^(n)(–2)" , "g^(n)(–2) \ n" } } ]
```

n	$f^{(n)}(-2)$	$g^{(n)}(-2)$
1	-0.109259118	-0.109259118
2	0.02410048667	0.02410048667
3	0.2551495138	0.2551495138
4	0.3716766307	0.3716766307

Bibliography

[1]Robert J. Lopez. **Advanced Engineering Mathematics**. Addison-Wesley. Boston. 2001.

[2]Richard L. Burden and J. Douglas Faires. **Numerical Analysis**, 7th edition. Brooks/Cole. NY. 2001.

SHIRLEY POMERANZ

Shirley Pomeranz is an Associate Professor of Mathematics in the Department of Mathematical and Computer Sciences at The University of Tulsa. She is the 2001-2002 ASEE Mathematics Division Chair and is a member of the Editorial Advisory Board for *The International Journal of Engineering Education*.

Her interests include support of women in mathematics and research involving the finite element method.