



*17th Annual International Conference on
Technology in Collegiate Mathematics
New Orleans, 2004*

ANIMATIONS USING MAPLE

Bill Blyth



Melbourne, Australia

**Department of Mathematics and Statistics
RMIT University, GPO Box 2476V, Melbourne, Australia 3001
bill.blyth@rmit.edu.au**

Abstract

We use the visualization and animation capabilities of Maple throughout our courses. Animations are used in presentations in class from first year onwards. For example, animations for Newton's method and for fixed point iteration; to illustrate how slicing is used to generate the domain for double integration; to show how the error function (for the collocation method solution of a boundary value problem) varies as the collocation points are changed.

This paper shows how to provide dynamically labelled animations. We start with a simple animation of parameterized curves where the value of the parameter is displayed on each frame. Our students in their first semester of first year asked to work individually or in pairs to attempt an animation assignment. The students are required to work in a group of size 3 plus or minus 1 and choose one of the 5 problems that are given. These problems are based on parameterized curves such as those that appear when modelling drug concentration in the blood. The presentation (in the lab) and assessment of each group's animation almost becomes a tutorial. This activity (which does not require calculus) is popular.

For our second example, we'll consider the standard calculus problem of finding the dimensions of the rectangle of maximum area that is inscribed in a triangle. Animation will be used to find the solution in a particular case. This solution can easily be shown to be the correct solution for the arbitrary case – no calculus is needed, only some simple algebra.

Finally, we briefly consider an advanced topic. From our third year Finite Element Methods FEM course, we briefly present an animation of the collocation method error function used to find very simply the “optimal” collocation points (a very difficult task analytically).

Introduction

Visualization has a long history (from the ancient Greeks) of effective use in mathematics. However the capability of modern computer software and calculators to provide visualization and animation provides new challenges to modify our curriculum and pedagogy to take advantage of these tools. Visualization in the teaching of mathematics was emphasized by the Harvard Consortium (supported by a NSF grant) which introduced and popularised the "rule of three": Every topic should be presented geometrically, numerically and algebraically, see [8].

In our courses, the visualization and animation capabilities of Maple are utilized. For example, Maple is used in a support mode where animations are presented students to assist with conceptual development in the case of ill-conditioning, see [7]. In a few more advanced courses, we use a “Maple immersion” mode: all teaching and assessment is done with Maple. For further general discussions and references; as well as details about a calculus of vector functions course in Maple immersion mode, see [3, 5]. In particular, in these papers, we discuss the use of the visualization capabilities of Maple to introduce the slicing idea in simple surface plots and to use the slicing diagrams of the domain of integration to correctly conceptualize double integration (as iterated or repeated integration). Animation is also used to illustrate the generation of the domain by slicing. The volume under a surface is treated as volume of a "shed". Line integrals are introduced to find the area of walls built above 2D paths. Slicing diagrams of 3D domains are also used to introduce triple integrals.

Our Geometry of Surfaces course is another example of an advanced Maple immersion mode course. This is some classical differential geometry, so it is characterized by a lot of symbolic (exact) computation and visualization – with very little numerical evaluation, see [4].

Introduction to Animation

In their first semester of first year, our students study Nonlinear Mathematics where Maple is used intimately in the course. Animations are used in class presentations for topics such as Newton’s method and fixed point iteration. Students also work through a worksheet on how to construct animations and are then required to complete an animation assignment [1]. Animation is introduced in the context of visualizing why the observed rate of convergence of Newton’s method is different for different values of the parameter in a parameterized family of functions, $2x^2 - \sqrt{x} + c$. It is useful to label each frame of the animation with the value of the parameter.

The code for dynamically labelled animations is

```
> restart;  
> with(plots):  
> picfn:= cc->plot(2*x^2 - sqrt(x) + cc, x=0..1,  
                  title=cat("c = ",convert(cc,string) ) );  
> display( [seq(picfn(c/16), c=0..16)], insequence=true);
```

This produces the animation ... of 17 frames. This can't be reproduced in a static document, so, in Figure 1, we display an array of 4 of the frames.

Note that the animation can be run and the speed changed, and it can be shown frame by frame. The interest here is that to notice the special value of c for which the graph just touches the x axis. This value is one of the c values for the animation, but is not one of the cases displayed in the array of frames in Figure 1. In the original context of this problem, it is now clear to the students that the case of $c = 1/4$ gives a simple zero of the function and so Newton's method rate of convergence is quadratic. However $c = 3/8$ is a special case where there is a double zero and so Newton's method converges linearly (i.e. slowly!).

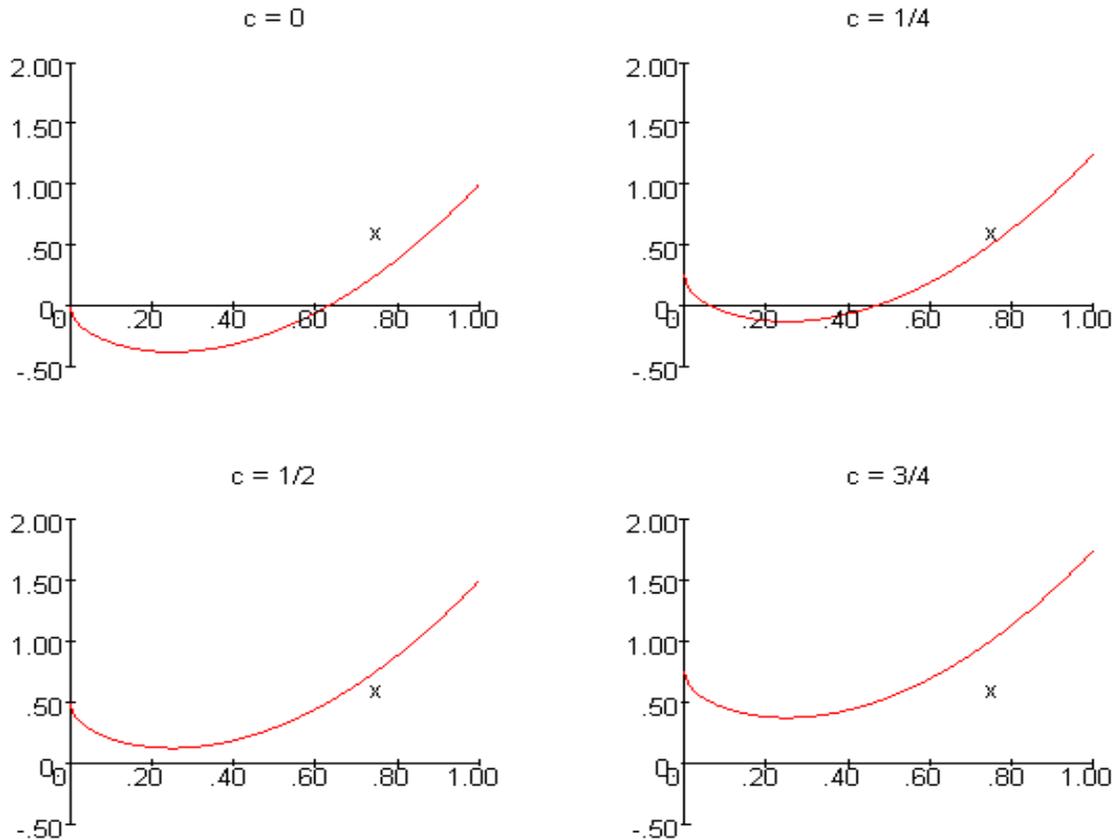


Figure 1. Four frames of the dynamically labelled animation.

The rectangle in a triangle problem

A rectangle is to be inscribed in a right triangle having sides of length 6 m and 8 m as shown in Figure 2. What are the dimensions of the rectangle having the maximum area? We first prepare an animation (not illustrated, although the code is given below) of the rectangle as its dimensions change and use `textplot` to display the values of the dimensions of the rectangle and its area. This shows the geometry very well. However we produce a further extended animation where the usual plot of area as a function of base of the rectangle is superimposed on the geometric plot. This is very effective conceptually, see Figure 3.

Animation can be used to find the solution (by experiment) in a particular case (and for further particular cases). This solution can easily be shown to be the correct solution for the arbitrary case – no calculus is needed, only some simple algebra.

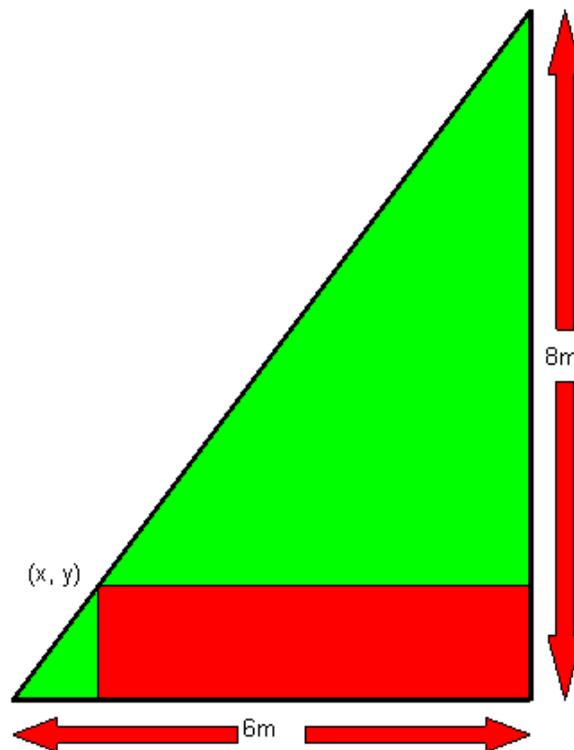


Figure 2. The rectangle inscribed in a triangle problem.

The code for the rectangle in a triangle problem:

```
> restart;  
> with(plots): with(plottools):  
> tri:=polygon([[0,0],[6,0],[6,8]], color=green,  
thickness=3):
```

This green triangle is defined here and used in the procedure as a global variable. For the animations, we also define the number of rectangles (of non zero area) as `n`.

```

>n:=20:      # 0<= i <=n
>picfn:=proc(ii)
  local xi, area, rect, tA;
  xi:=ii*6/n;
  area:=4/3*xi*(6-xi);
  rect:=rectangle([xi,4/3*xi], [6,0], color=red);
  tA:=textplot([
  [.5,8.8,cat("Width = ",convert(evalf(6-xi),string))],
  [.5,8,cat("Height= ",convert(evalf(4/3*xi),string))],
  [.5,7.2,cat("Area  = ",convert(evalf(area),string))]],
  align=RIGHT, color=red);
  display(rect, tri, tA);
end:
>display(seq(picfn(i),i=0..n), insequence=true);

```

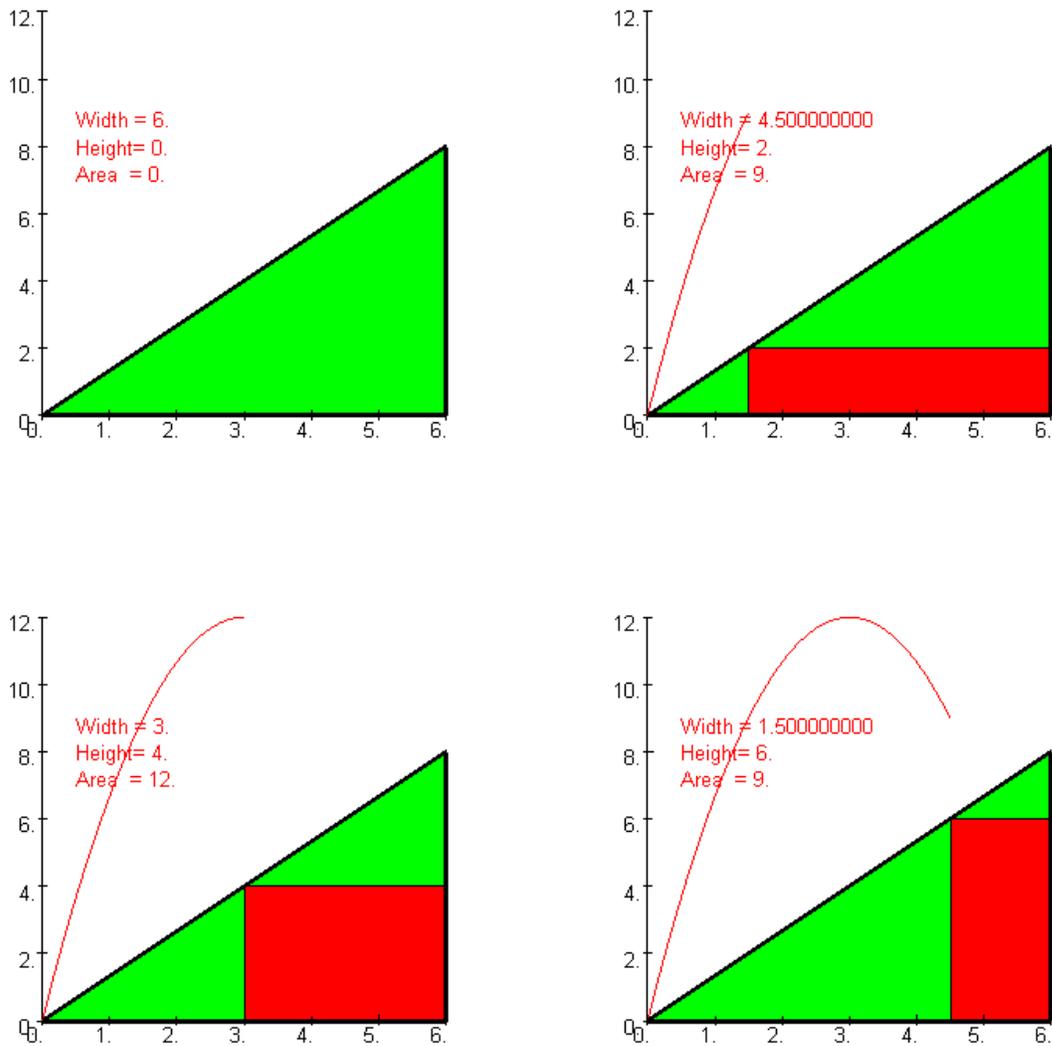


Figure 3. Four frames of the rectangle in a triangle “extended” animation.

The additional code for the extended animation for the rectangle in a triangle problem:

```
>picfn2:=proc(ii) # ii > 0
  local xi, xx, area, rect, tA, curve;
  xi:=ii*6/n;
  area:=4/3*xi*(6-xi);
  rect:=rectangle([xi,4/3*xi], [6,0], color=red);
  tA:=textplot([
  [.5,8.8,cat("Width = ", convert(evalf(6-xi),string))],
  [.5,8,cat("Height= ", convert(evalf(4/3*xi),string))],
  [.5,7.2,cat("Area = ",convert(evalf(area),string))],
  align=RIGHT,color=red);
  curve:=plot(4/3*xx*(6-xx), xx=0..xi, view=[0..6,0..12],
  labels=["", ""]);
  display(rect, tri, tA, curve);
end:
>display(picfn(0), seq(picfn2(i),i=1..n), insequence=true);
```

Animation of error as collocation points vary

The test problem that is used to introduce the collocation method is as follows.

Solve the two-point boundary value problem

$$u'' + u = -x$$

for $0 < x < 1$ subject to the boundary conditions

$$u(0) = 1 \text{ and } u'(1) = -1$$

using the collocation method.

Cubic polynomials are chosen as the trial functions, u , and the boundary conditions are imposed on u . The residual is calculated by substituting this trial function into the left hand side of the differential equation minus the right hand side. The residual is set equal to zero at 2 collocation points (at $x = \text{collPt}$ and $1 - \text{collPt}$). The error (function) varies according to the choice of the collocation points. It is very difficult to determine the “optimal” choice of the collocation points. However we can illustrate how the collocation method error varies (as the collocation points vary) and determine the (near) optimal choice of the collocation points by using the animation illustrated in Figure 4. From the animation, the best choice for the collocation points is $x = 7/32$ and $1 - 7/32$.

For a more accurate determination of the optimal choice of the collocation points, we could re-run the animation (not illustrated here) having “zoomed in” on the interval between $6/32$ and $8/32$, with steps of $1/8$ of the previous size $1/32$ (i.e step size = $1/256$). This still shows that $x = 7/32$ (and the symmetric point $1 - 7/32$) is the best choice.

The error, with cubic polynomials and 2 collocation points at collPt and $1 - \text{collPt}$, is

$$err := collPt \rightarrow \cos(x) + \frac{\sin(1) \sin(x)}{\cos(1)} + \frac{(collPt^2 - collPt + 4)(-2x + x^2)}{collPt^4 - 2 collPt^3 + 8 collPt^2 - 7 collPt + 8} + \frac{-3x + x^3}{collPt^4 - 2 collPt^3 + 8 collPt^2 - 7 collPt + 8} - 1$$

The code for the collocation method animation: first enter the formula for the err (given in output form above) and then the following.

```
> with(plots): with(plottools):
> ptPlot:=pt->scale(disk(pt,0.01,color=blue),1,1/3); #pt is [x,y]
> picfn:=proc(cp)
local collocationPts, errCurve;
collocationPts:=[[cp,0],[1-cp,0]];
errCurve:=plot(err(cp), x=0..1,
  title=cat("coll pts = ",convert(cp,string)," & ",
  convert(1-cp,string)), titlefont=[TIMES, ROMAN,16]);
display(errCurve,map(ptPlot,collocationPts));
end:
> display( [seq(picfn(cp/32), cp=1..15)], insequence=true);
```

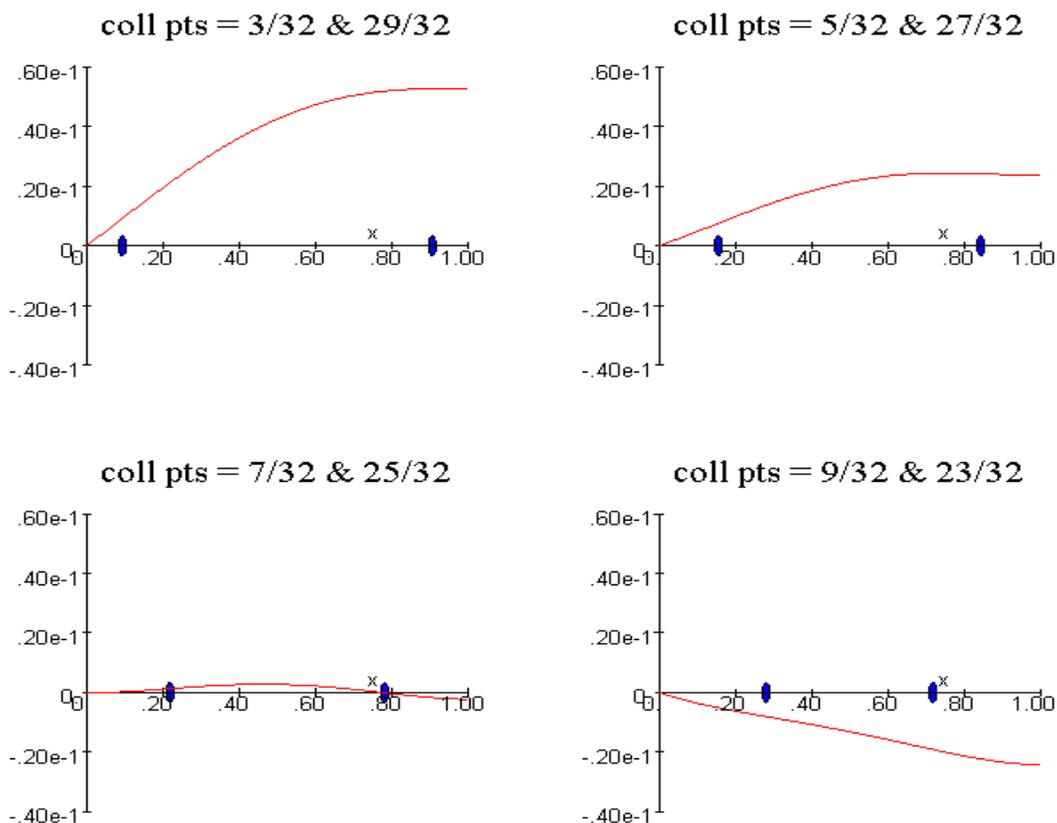


Figure 4. Four frames of the collocation method animation.

Conclusion

Modern computer software such as Maple and Mathematica can produce high quality and quite complicated visualizations and animations. We have discussed animations used in innovative ways in our teaching of undergraduate mathematics. Our experience is that students respond well and enjoy class presentations and laboratory work that involves visualization and animation. The cost of staff resources to develop these visualization and animation teaching materials is high – our rule of thumb is that one hour in the clsas or the lab takes about 10 hours to prepare. However the educational return is also high – we will continue to seek opportunities to incorporate visualization and animation into our teaching.

References

- [1] Blyth, B., Animations using Maple in First Year, *Quaestiones Mathematicae*, Suppl., 1 , Supplement, 201--208, 2001.
- [2] Blyth, B., Finite Element Methods: Presentation and Animation using Maple, *Vienna International Symposium on Integrating Technology into Mathematics Education (VISIT-ME-2002)*, Proc., J.Boehm, editor, bk teachware, Austria, (CDROM), 2002.
- [3] Blyth, B., Visualization of Slicing Diagrams using Maple, *EMAC 2003 Proceedings*, University of Technology, Sydney, editors R.L. May and W.F. Blyth, Engineering Mathematics Group, ANZIAM, Australia, 2003, 7-12.
- [4] Blyth, B., Geometry of surfaces using Maple, *New Zealand Journal of Mathematics*, 32 Supplementary Issue, 29--36, 2003.
- [5] Blyth, B., Multiple Integration - Visualization and Animation using Maple, *Proceedings of the 17th International Conference on Technology in Collegiate Mathematics: ICTCM (2004)*, New Orleans, USA, to appear.
- [6] Blyth, B. and Naim A., Problem Solving using Maple Animations, *Proceedings of the 6th Asian Technology Conference in Mathematics: ATCM 2001*, published by ATCM Inc, USA, 273--281, held at RMIT, December 2001.
- [7] Blyth, B., Vouzas, P. and Saunders, J., Animations to illustrate ill-conditioning and an introduction to matrices using Maple, in *Proc of the 2nd International Conference on the Teaching of Mathematics: ICTM2*, University of Crete, Greece, editor M. Boezi, CD published by John Wiley, ISBN 0-471-46332-9, 2002.
- [8] Hughes-Hallett, D., Gleason, A.M., et al., *Calculus*, John Wiley, 1994.
- [9] Putz, J.F., *Maple Animation*, Chapman & Hall/CRC, 2003.