A FOUR STEP METHOD FOR CREATING ANIMATIONS

Rosemary Carroll Farley, Ph. D.
Patrice Geary Tiffany, Ed. D.
Manhattan College

Abstract

This paper describes a four step method to create complicated graphs and animations using the computer algebra system Maple. This four step method has been successfully used with undergraduate students to encourage mathematical research. This paper will demonstrate how easy it is to adapt this method to create new and vastly different graphs and animations.

Introduction

This paper describes a four step method to create complicated graphs and animations using the computer algebra system Maple without the need for a great deal of coding. This paper was motivated by a desire to involve students in mathematical research. Often students can be convinced to start mathematical experimentation using the power of a computer algebra system. The four step method has been used with great success with students who are trying to create their own graphs and animations. Whenever a student is thinking about a research question and graphing is involved, the student is given a description of this four step method to read. The goal in this paper is to show how easy it is to adapt this method to create new and vastly different graphs and animations. Each of the following examples came from student research questions.

The Four Step Method

Step One: Plot one component of the animation.
Step Two: Create and name the plots.
Step Three: Create a list of these plots.
Step Four: Display the animation.

Example 1: The Orbit of a Complex Number

The orbit of a complex number z is the set O(z) = {z, z^2, … z^k , …}. The four step method will be used to create an animation of the orbit of any complex number. First load the appropriate libraries.

$with(plots) : with(linalg) :$

Step One: Plot one component of the animation. In this example, a complex number z is chosen and plotted.

$$z := \frac{\mathrm{sqrt}(2)}{2} + \frac{\mathrm{sqrt}(2)}{2} \cdot I :$$

$$complexplot(z, x = -10 .. 10, style = point, symbolsize = 30)$$

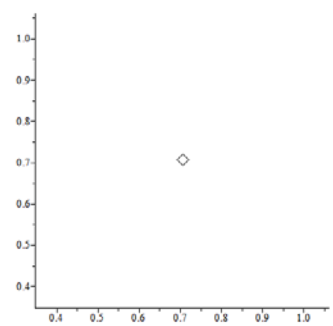

Figure 1:

Step 2: Create and name the plots. Modify step 1 to account for what is changing in the animation. Be sure to give each plot a different name as follows.

for n from 1 to 40 do
$Q[n] := complexplot(z^n, x = -10 .. 10, style = point, symbolsize = 30)$
od:

Once step two is completed, the next two steps need only be modified by the number of frames in the animation.

Step 3: Create a list of plots.

$L := [\,seq(Q[n], n = 1..40)\,]:$

Step 4: Display the animation. Because this is a print medium, every point in the animation will be displayed. To see the animation, change the word false to true. Press the play button that appears when the graph is highlighted.

$display(L, insequence = false)$

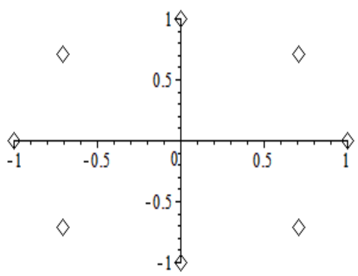

Figure 2:

Suppose that the unit circle is to be a permanent part of the animation. Give the above display a name.

$d1 := display(L, insequence = false):$

Create and name the plot of the unit circle on the complex plane and display both plots together.

$p1 := complexplot(\cos + \sin \cdot I, -\mathrm{Pi}..\mathrm{Pi}):$
$display(\{d1, p1\})$

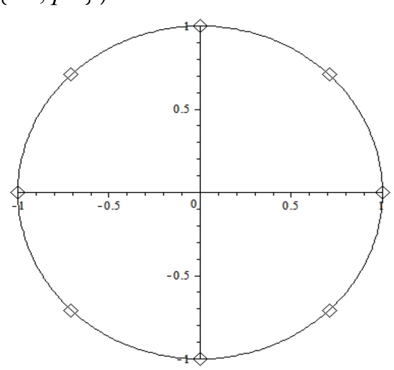

Figure 3:

Example 2: The Orbit of a Linear Transformation

Suppose that T is a linear transformation from R^2 to R^2 and A is the 2 x 2 matrix associated with the transformation. Let x_0 be an arbitrary point in R^2. Let $x_n = Ax_{n-1}$. The set $\{x_0, x_1, x_2, x_3, \ldots\}$ is called the orbit of the transformation T for the point x_0. Another way to write the orbit is $\{x_0, Ax_0, A^2 x_0, A^3 x_0, \ldots\}$. The four step method is used to create an animation of the orbit of any linear transformation from R^2 to R^2. Select a 2 x 2 matrix A and a point x in R^2.

$$A := \begin{bmatrix} 0.5 & -0.6 \\ 0.75 & 1.1 \end{bmatrix} : x := \begin{bmatrix} 2 \\ 1 \end{bmatrix} :$$

Step One: Plot one component of the animation. In this example, a point in R^2 is plotted.

$pointplot(x, style = point)$

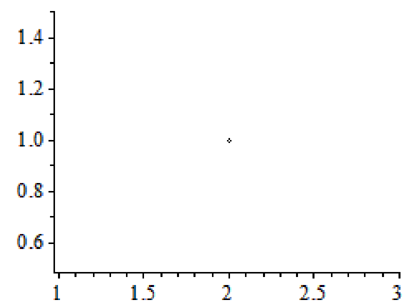

Figure 4:

Step Two: Create and name the plots. In this example, Q[1] is the plot of x. The matrix A is multiplied by x to find the second point in the orbit. Q[2] is the plot of that point. This process continues inside the loop until Q[80] is created.

```
for n from 1 to 80 do
    Q[n] := pointplot(x, style = point) :
    x := multiply(A, x) :
od:
```

Step Two is the difficult step. Now the next two steps need only be modified by the number of frames in the animation.

Step Three: Create a list of these plots.

```
L := [seq(Q[n], n = 1 ..80)] :
```

Step Four: Display the animation. Once again, every point in the animation will be displayed below.

```
display(L, insequence = false, axes = normal)
```

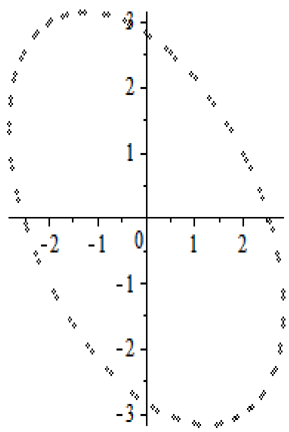

Figure 5:

Example 3: Point Plotting From A Spreadsheet

A three dimensional plot was created using points listed in an Excel spreadsheet. The first column of the spreadsheet contained twenty one different x values, the first row contained eight different y values, and corresponding z values had been put into the appropriate row and column. Cell A1 was empty. The data was imported into Maple. The four step method was then used to create the three dimensional plot. First, load the appropriate libraries.

```
with(plots) : with(Spread) :
```

The file is imported into the Maple worksheet by choosing "Insert" then "Spreadsheet" and giving the spreadsheet a name. With the cursor in the spreadsheet, choose "Import" then "tab delimited" from the spreadsheet menu and pick the selected spreadsheet. The chosen spreadsheet has the name "Data." For the sake of space, the spreadsheet “Data” is omitted.

Step One: Plot one component of the animation. A three dimensional plot of a point is to be created. The correct code for the point and 3d plot follows.

```
P := [1.01, 0.01, 5]
pointplot3d(P, axes = boxed)
```

Step Two: Create and name the plots. This can be a difficult step. The x, y, and z values are to be read from the spreadsheet where x is the i^{th} value in the first column, y is the j^{th} value in the first row, and z is the value in the i^{th} row and j^{th} column. Counters are needed for both row and column. Nested for--do loops will be used. The command GetCellValue(A, i, j) will get the value in the i^{th} row and j^{th} column of the spreadsheet called A. Cell A1 is empty and the x values start in the second column and the y values start in the second row. Plots are created by putting everything inside nested for--do loops. The counter n is needed here because we need to give the plots different names.

```
n := 1 :
  for i from 2 to 22 do
    for j from 2 to 9 do
    P := [GetCellValue(Data, i, 1), GetCellValue(Data, 1, j),
     GetCellValue(Data, i, j)] :
    Q[n] := pointplot3d(P, symbol = point, symbolsize = 15, color
      = black) :
    n := n + 1 :
    od:
  od:
```

The difficult work is done. Now repeat steps three and four, changing only the number of frames. In the previous step, 21 * 8 = 168 plots were created.

Step Three: Create a list of these plots.

```
L := [seq(Q[n], n = 1 ..168)] :
```

Step Four: Display the graph. With no command regarding insequence, all plots will be shown simultaneously.

$display(L, axes = boxed)$

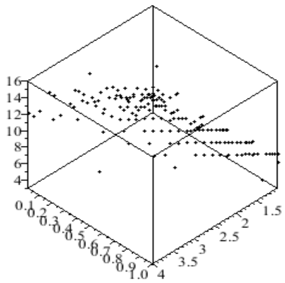

Figure 6:

Suppose that the plane z= 3 x + 2 y + 2 is to be added to the above display. Name the above display, create and name the 3 dimensional plot of the plane, and display them together.

$d1 := display(L, axes = boxed) :$
$p1 := plot3d(3 \cdot x + 2 \cdot y + 2, x = 0 ..5, y = 0 ..1) :$

$display(d1, p1)$

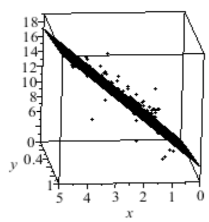

Figure 7:

CONCLUSION

This four step method can be used to create complicated graphs and animations without a great deal of coding. It is particularly useful for students who are initially getting involved with research. With a computer algebra system, students begin to experiment with ideas, to visualize results, and ultimately to form hypotheses to be proven or discarded. The four step method is a systematic way to simplify the creation of complicated animations and graphs. The method is successful because it is both easy to use and because it places the emphasis on what is important--the mathematics involved.

Biographical Information

Dr. Rosemary Farley is an Associate Professor of Mathematics at Manhattan College. Her research interest concerns the appropriate use of technology in the mathematics classroom. This includes analyzing the impact that the incorporation of a computer algebra system has in higher level collegiate mathematics courses and graphing calculator technology in courses for prospective mathematics teachers.

Dr. Patrice Geary Tiffany is an Associate Professor of Mathematics and Computer Science at Manhattan College. Her research interest concerns the infusion of technology in the Calculus curriculum with its effect on that curriculum. This technology includes computer algebra systems and present day calculators as well as online components such as text, homework and testing. Present research is also focusing on how online technology can best enable a successful “flipped” classroom.