Spreadsheets in Education (eJSiE)

Volume 2 | Issue 1 Article 8

1-5-2006

Enhancing Mathematical Graphical Displays in Excel through Animation

Deane Arganbright
Department of Mathematics Korea Advanced Institute of Science and Technology, dearganbright@yahoo.com

Follow this and additional works at: http://epublications.bond.edu.au/ejsie

Recommended Citation
Arganbright, Deane (2005) "Enhancing Mathematical Graphical Displays in Excel through Animation," *Spreadsheets in Education (eJSiE)*: Vol. 2: Iss. 1, Article 8.
Available at: http://epublications.bond.edu.au/ejsie/vol2/iss1/8

This In the Classroom Article is brought to you by the Faculty of Business at ePublications@bond. It has been accepted for inclusion in Spreadsheets in Education (eJSiE) by an authorized administrator of ePublications@bond. For more information, please contact Bond University's Repository Coordinator.

Enhancing Mathematical Graphical Displays in Excel through Animation

Abstract
Enhancing Mathematical Graphical Displays in Excel through AnimationThe use of spreadsheets in teaching mathematics has increased significantly in recent years. One can implement mathematical algorithms, models, visualizations, and applications naturally and effectively through interactive spreadsheet constructions and creative graphical displays. This paper demonstrates techniques that enable educators to design animated graphical displays in their spreadsheet constructions in order to produce powerful classroom demonstrations to enhance mathematical understanding, while also presenting students with new ideas for incorporating attractive visual components in their mathematical assignments and projects.

Keywords
spreadsheet, animation, model, graphing, visualization

This in the classroom article is available in Spreadsheets in Education (eJSiE): http://epublications.bond.edu.au/ejsie/vol2/iss1/8

Enhancing Mathematical Graphical Displays in *Excel* through Animation

Deane Arganbright
Department of Mathematics
Korea Advanced Institute of Science and Technology
arganbright@math.kaist.ac.kr

Abstract

The use of spreadsheets in teaching mathematics has increased significantly in recent years. One can implement mathematical algorithms, models, visualizations, and applications naturally and effectively through interactive spreadsheet constructions and creative graphical displays. This paper demonstrates techniques that enable educators to design animated graphical displays in their spreadsheet constructions in order to produce powerful classroom demonstrations to enhance mathematical understanding, while also presenting students with new ideas for incorporating attractive visual components in their mathematical assignments and projects.

Keywords: spreadsheet, animation, model, graphing, visualization.

1 Introduction

Almost every mathematics educator employs some form of visual representation to convey mathematical ideas and enhance the conceptualization and teaching of mathematics. For many years, these representations consisted primarily of textbook illustrations and sketches drawn by teachers on blackboards. However, textbook images are static, providing only frozen illustrative instants of a dynamic process. Hand drawn figures on the board do allow for some limited animation, primarily through the physical drawing process and subsequent gesturing by a teacher. However, the availability of interactive and animated visualizations afforded by modern technology can enhance the learning process significantly. Consequently, we see an increasing number of text supplements and Web sites [2], [3], [11] that provide animated graphic illustrations.

At the same time, a spreadsheet, such as *Microsoft Excel*, provides us with a natural, interactive medium for doing mathematics. Perhaps surprisingly, spreadsheets are also effective tools for designing mathematical animations. For example, instead of studying Newton's method through a formal mathematical development accompanied by a few fixed pictures and some laborious hand computations, we can implement the algorithm together with an animated graph in a spreadsheet [7]. We can then use the animation not only to view the typical rapid convergence of the algorithm, but also to discover some surprising instances of divergence or of convergence to an unexpected zero.

We have designed this paper to provide teachers and students with tools that they can use to create their own effective animations through *Microsoft Excel*, which is perhaps the principal mathematical tool of the workplace. It is also an excellent tool for mathematics instruction, mathematical modeling, experimentation by students, and visualization. This article illustrates how we can implement some of the primary spread-

© 2005 Spreadsheets in Education, Bond University. All rights reserved.

sheet visualization techniques in creating effective interactive and animated graphic educational models, especially in mathematically oriented areas. In addition to describing these techniques, we provide the actual *Excel* files for our examples so that readers can see the animation in action, study the details of the implementations, and acquire creative skills. We create all of our examples using only standard *Excel* with no add-ins. While we use *Excel 2003* in *Windows XP* to create the accompanying spreadsheet files, one can also create these models using a *Macintosh* version of *Excel* with only minor adaptations required.

The examples presented represent a small illustrative sample of what can be accomplished in animating spreadsheet graphics. It is possible to do much more, including creating animated visualizations of mathematical models [7]. A list of references provides a way to locate additional material. We can use these examples for classroom demonstrations, as illustrative prototypes of student projects, and as a springboard for personal development activities for professional educators. At the end of the paper, we provide suggestions for additional projects.

It is possible to create effective animated visualizations for the great majority of topics encountered in such fundamental undergraduate mathematics classes as algebra, pre-calculus, calculus, linear algebra, differential equations, statistics, operations research, and numerical analysis by producing animated versions of the examples and illustrations contained in textbooks. We can even create 3-dimensional illustrations of many topics by implementing the perspective drawings found in texts.

These capabilities allow classroom teachers to create eye-catching demonstrations for such topics as constructing curves and tangent lines, visually investigating the effects of changes in initial values and parameters of such algorithms as Newton's and Euler's methods, and interrogating a wide range of mathematical models. Teachers and students can create spreadsheet models illustrating such diverse topics as approximating the area under a curve by a varying number of rectangles or graphically solving a 2-variable linear program by moving the graph of the objective function in a continuous manner via a scroll bar. Frequently we can even motivate mathematical proofs in this manner, as we illustrate in this paper.

Students often find that incorporating animation effects in their assignments not only presents them with intriguing opportunities for designing attractive output, but also introduces them to new mathematical concepts and challenges along the way. The author has used spreadsheet animations in teaching virtually all undergraduate mathematics courses, as well as in graduate classes for in-service secondary mathematics teachers, where the approach has been effective in generating new insights and interest among teachers, while providing examples that they can incorporate into their own teaching. The use of a spreadsheet has been especially effective in teaching in a developing nation, where other mathematical software is not readily accessible.

2 Animating function graphs

One of the principal topics that we consider early in the study of calculus is the use the first and second derivatives in analyzing functions and their graphs. In our first example, we present a way to create interactive, animated graphs of a function and its first two derivatives, rather than just producing motionless pictures. We create the graph of a quartic polynomial, $f(x) = ax^4 + bx^3 + cx^2 + dx + e$, where the coefficients are entered as parameters. This model is designed for classroom demonstrations. However, students in

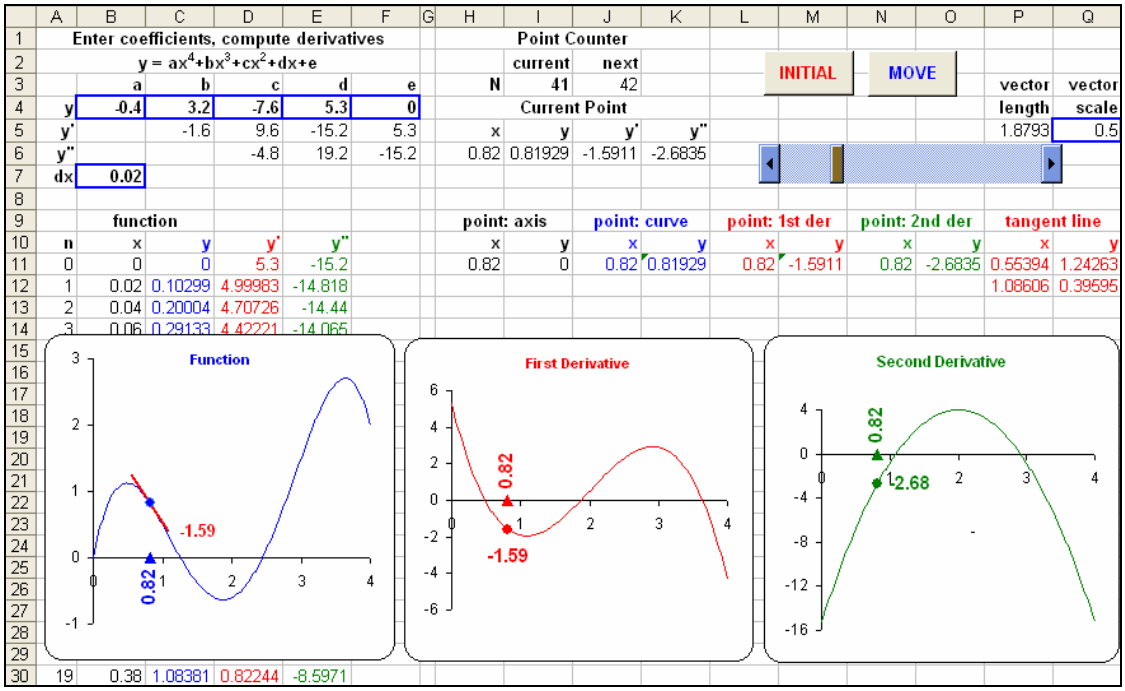


Figure 1: Graphing model

algebra, pre-calculus, and calculus can incorporate many of its features into assignments or projects. Figure 1 shows the *Excel* display of our spreadsheet construction.

2.1 Arrow formula diagrams

Throughout this paper, we will outline the cells into which we enter values. Thus, we enter the function's coefficients in Row 4, while the formulas in the next two rows compute the coefficients of the function's first and second derivatives. In Figure 2 we use the arrow notation introduced in [7] and [8] to describe our spreadsheet formulas. Thus, a dot indicates a cell that a formula references, while an arrow points to the location within the formula where the value of the referenced cell is used. This notation allows us to describe our model without the need to refer to specific physical cell locations. For example, the coefficient of x^3 in the first derivative is 4 times the coefficient of x^4 in the original function. We employ variously colored lines and markers to assist us in interpreting the diagrams.

In Figure 2 we use a light shading to indicate that we generate the values for the second derivative by copying formulas in the dark shaded cells down the respective columns. Thus, the coefficient of x^2 in the first derivative is $3b$, while the coefficient of x^2 in the second derivative is $3(4a)$. The ability to copy the formulas for successive derivatives in this manner essentially allows us to implement differentiation as a mathematical operator. The cell references in Figure 2 are relative references (*i.e.* each represents "the cell above and to the left"). A spreadsheet adjusts relative references to preserve their positional relationships in the copying process. In Figure 3 we provide the traditional notation for this construction using *Excel* formulas.

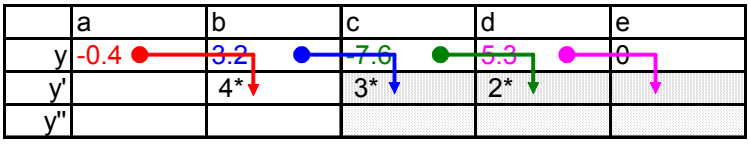


Figure 2: Derivative formulas via arrow diagrams

	A	B	C	D	E	F
3		a	b	c	d	e
4	y	-0.4	3.2	-7.6	5.3	0
5	y'		=4*B4	=3*C4	=2*D4	=E4
6	y''			=3*C5	=2*D5	=E5

Figure 3: Derivative formulas via *Excel* notation

2.2 Graphing

To create our graphs, we use the first five columns to generate the coordinates of points on the function together with the corresponding derivatives at these points. As described in Figure 4, the first column serves as a counter, $0 \le n \le 200$, for the points for the graph. The second column generates successive values of x, starting with 0, and incremented in steps of size dx, where we enter the value of dx in the indicated cell above. In addition, the third column computes the corresponding values of y using the coefficients. The dots in the coefficient cells show pins stuck through them, indicating they are absolute references that will remain fixed in copying. We compute the values for y' and y'' in a like manner. After entering the initial expressions, we copy them down their respective columns.

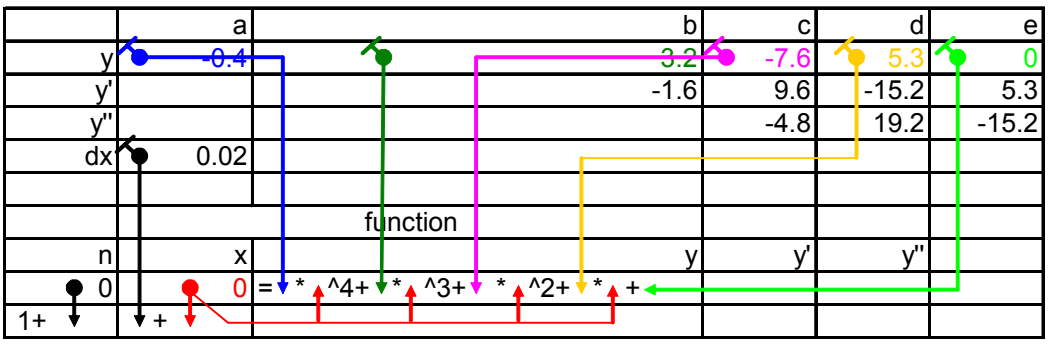


Figure 4: Formulas for n, x, and y via arrow diagram

For comparison, in Figure 5 we show the same formulas in standard *Excel* format. References indicated by $ in *Excel* (or with a pin in the arrow notation) are absolute, while those without the $ (or plain dots in the arrow notation) are relative references. This table shows one reason why using the arrow notation makes working with textual descriptions much easier than the standard formula representation.

	A	B	C
10	n	x	y
11	0	0	=$B$4*B11^4+$C$4*B11^3+$D$4*B11^2+$E$4*B11+$F$4
12	=1+A11	=B11+$B$7	=$B$4*B12^4+$C$4*B12^3+$D$4*B12^2+$E$4*B12+$F$4
13	=1+A12	=B12+$B$7	=$B$4*B13^4+$C$4*B13^3+$D$4*B13^2+$E$4*B13+$F$4

Figure 5: Formulas for n, x, and y in *Excel* notation

To create the graph of the original function from these values, we highlight the x- and y-columns (Fig. 6) and press the Chart Wizard button. In the ensuing series of dialog boxes, we choose the xy-chart type, displaying both markers and lines. This graph type plots points in a Cartesian grid by their (x, y) coordinates, and connects consecutive points with line segments. We include markers since this makes it easier for us subsequently to insert a graph series that consists of a single point. Later we can eliminate the markers or the lines if desired.

We use similar techniques to plot a separate graph of the first derivative (Fig. 7), by first highlighting the x column, holding down on the Ctrl key, selecting the y' column, and using the Chart Wizard. We also use the same process in plotting a graph of the second derivative.

9		function			
10	n	x	y	y'	y''
11	0	0	0	5.3	-15.2
12	1	0.02	0.10299	4.99983	-14.818
13	2	0.04	0.20004	4.70726	-14.44
14	3	0.06	0.29133	4.42221	-14.065
15	4	0.08	0.37698	4.14462	-13.695
16	5	0.1	0.45716	3.8744	-13.328
17	6	0.12	0.53201	3.61148	-12.965
18	7	0.14	0.60167	3.35577	-12.606

Figure 6: Graphing $f(x)$

9		function			
10	n	x	y	y'	y''
11	0	0	0	5.3	-15.2
12	1	0.02	0.10299	4.99983	-14.818
13	2	0.04	0.20004	4.70726	-14.44
14	3	0.06	0.29133	4.42221	-14.065
15	4	0.08	0.37698	4.14462	-13.695
16	5	0.1	0.45716	3.8744	-13.328
17	6	0.12	0.53201	3.61148	-12.965
18	7	0.14	0.60167	3.35577	-12.606

Figure 7: Graphing $f'(x)$

Next, we embellish our graphs by incorporating a trace feature and a tangent line. To do this we set aside one cell (here Cell I3) to use as a point counter, N. We then reproduce the various values for point N by using *Excel*'s offset function. We show the expression for x in Figures 8 and 9. The offset function, whose *Excel* formula is =OFFSET(B11,$I$3,0), starts from Cell B11 and returns the value that is offset from that cell by N rows and 0 columns. We then copy the formula to the right to obtain the other values of point N.

N	4		
Current Point			
x	y	y'	y''
0.08	0.37698	4.14462	-13.695

Figure 8: Values of current point, N

a	b	c	d	e		N	4		
-0.4	3.2	-7.6	5.3	0		Current Point			
	-1.6	9.6	-15.2	5.3		x	y	y'	y"
		-4.8	19.2	-15.2		OFFSET(, ,0)			
0.02									
function									
x	y	y'	y"						
0	0	5.3	-15.2						

Figure 9: Formulas for current point, *N*

We now use several columns of the spreadsheet display to generate the coordinates for the points $(x,0)$, $(x,f(x))$, $(x,f'(x))$, and $(x,f''(x))$. We describe the constructions of $(x,0)$ and $(x,f(x))$ in Figures 10 and 11.

point: axis		point: curve	
x	y	x	y
0.08	0	0.08	0.37698

Figure 10: Two points (output)

x	y	y'	y"
0.08	0.38	4.14	-13.69
point: axis		point: curve	
x	y	x	y
	0		

Figure 11: Two points (formulas)

To include these points in our graph, we highlight the appropriate (x,y) cells with the mouse, hold down on the left mouse button, drag the block into the various graphs, and release the button (see Fig. 12). In the resulting dialog box (Fig. 13), we indicate that this is a new series with the *x*-coordinates in the first column. We can enhance the usefulness of our graphs by using labels for points and lines. We describe the implementations of such additional features within the accompanying *Excel* file, Example 5.

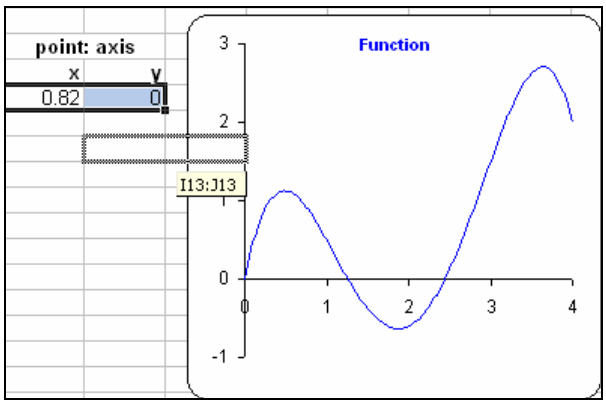


Figure 12: Dragging a new series into a graph

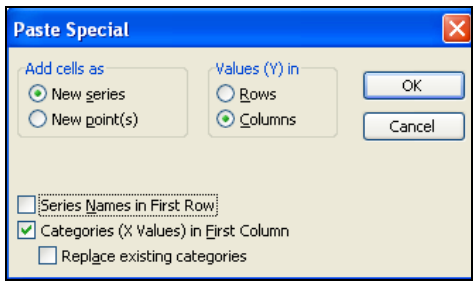


Figure 13: New series box

In the graph of the original function, we also include a tangent line at the trace point. To create this line, we observe that the vector $[1,y']$ is tangent to the curve and its length is $\lambda=\sqrt{1+y'^2}$. Thus $[1,y']/\lambda$ is a unit tangent vector. We enter a reasonable scale factor, *s*, to create a tangent vector at (x,y) by using the points $(x\pm s/\lambda, y\pm sy'/\lambda)$. Again, we highlight the block of these points and drag it into the graph of the function.

2.3 Scroll bars

Having created our model, there are two primary ways to incorporate animation to move the trace point and tangent line. The first is by using a scroll bar, or slider. The PC version of *Excel* has two different scroll bars. We use the one on the Control Toolbar, since it will create the smoothest animation effect (on the *Macintosh* version of *Excel*, the only scroll bar is on the Forms toolbar). To generate a scroll bar to link to the value of *N*, we first select from the command menu: View, Toolbars, Control Toolbox (Fig. 14).

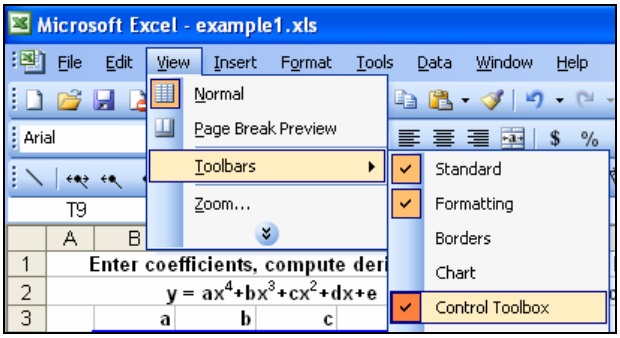


Figure 14: Selecting control toolbox

Next, we toggle the Design Mode button in the upper left corner of the toolbar, and then select the scroll bar icon (Fig. 15). We then use the mouse to drag out an image of the scroll bar and release. We next right click in the resulting scroll bar and choose the Properties option (Fig. 16), setting the Linked Cell to be the cell that contains the value of *N* (here Cell I3), and enter Minimum and Maximum values for *N* (here 0 and 200) (Fig. 17). We then toggle the Design Mode button to exit the design mode. Now the scroll bar is active. As we move its slider, the value of *N* changes. This in turn causes the trace point linked to *N* to move within the graph.

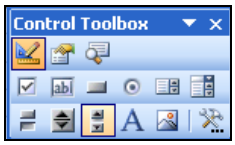


Figure 15: Control toolbox

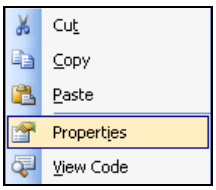


Figure 16: Toolbox options

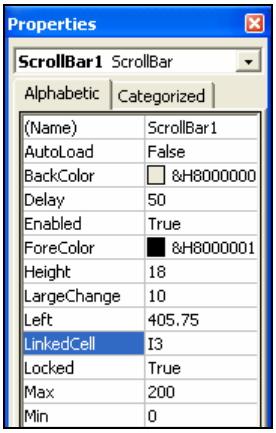


Figure 17: Scroll bar properties

The scroll bar allows us to vary *N* in an essentially continuous manner and thereby to examine the resulting effects. If we find an interesting location, we can pause and move the trace point in either direction to investigate further.

2.4 Creating movie clips through macros

For a second animation approach, we can create a "movie" clip to show the complete tracing of the point in a continuous fashion. To do this, we employ a cell adjacent to that of the current counter value, *N*, to contain a formula to compute the next counter value, *N*+1, as illustrated in Figures 18 and 19.

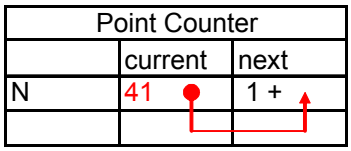


Figure 18: Updating *N* (formula)

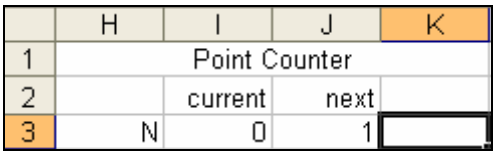


Figure 19: Updating *N* (output)

We will use this construction as a way to increase continually the value of *N* through the creation of a macro – a small program written using *Visual Basic*. Fortunately, we do not need to know much about programming in this language, because we can generate a macro by turning on *Excel*'s macro recording feature. Thus, we click in Cell K3 (Fig. 19) before we start recording. Next, we issue the command Tools, Macro, Record New Macro (Fig. 20). In the ensuing dialog box (Fig. 21) we give the macro a name, "Move".

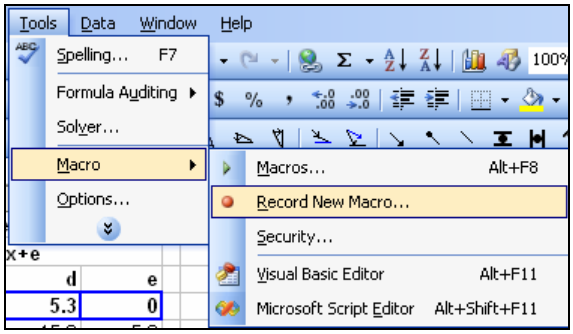


Figure 20: Recording a macro

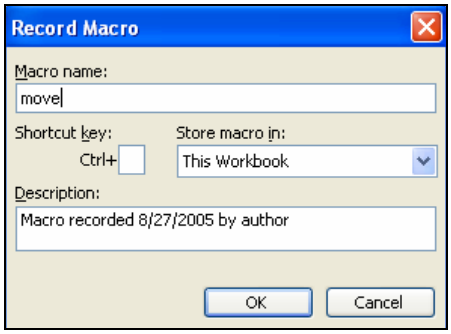


Figure 21: Naming a macro

We now click on the "next value" Cell J3, then right click, and choose the option Copy (Fig. 22). We then click in the "current value" Cell I3, and chose the option Edit, Paste Special (Fig. 23), and choose Values in the resulting dialog box (Fig. 24). This increases the values of both *N* and *N*+1 by 1.

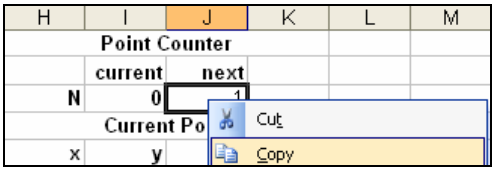


Figure 22: Macro step (copy)

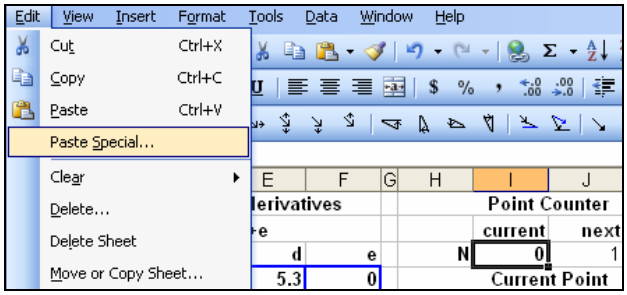


Figure 23: Macro step (paste special)

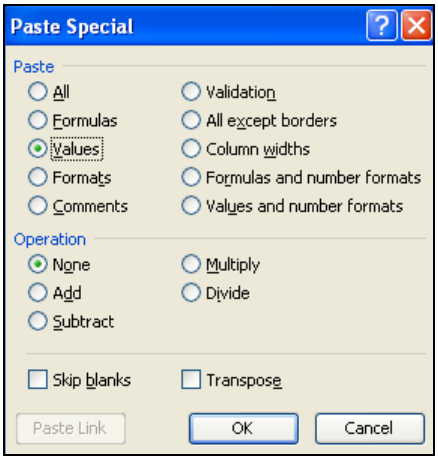


Figure 24: Macro selection (values)

We now click back to Cell K3 and press the Escape button in order to remove the dotted lines that will have appeared around Cell I3. Finally, we click on the End Record button that appears when we record macros, or issue the command Tools, Macro, Stop Recording. We now have created a macro that will increase the value of *N* by 1 when we issue the command to run the macro. However, to create the desired animation we want to have this done continuously 200 times. Consequently, we must modify our macro. To do this we enter the commands Tool, Macros, select the macro Move, and click the option Edit. There we will see the *Visual Basic* macro that we have just recorded. We list the macro below.

```
Sub move()
' move Macro
' Macro recorded 8/27/2005 by author
   Range("J3").Select
   Selection.Copy
   Range("I3").Select
   Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
      :=False, Transpose:=False
   Range("K3").Select
   Application.CutCopyMode = False
End Sub
```

We want to modify this macro in two ways. First, we enclose the macro inside a loop by typing in bracketing expressions "For *I* =1 to 200" at the top, and "Next *I*" at the bottom. Now when the Move macro is run, the loop is repeated 200 times with *I* increasing from 1 to 200 in steps of size 1. In the graph, we will see the trace components moving continuously.

However, sometimes the resulting motion may proceed faster than we desire. In this case, we can slow the looping by inserting an extra time-consuming operation in each pass. Here we insert another loop that adds the integers 1 to 100,000 and stores the results in the variable Sum each time through the loop. We will never use the resulting sum, but the computation will slow the macro. When we have finished, we exit the macro edit mode by pressing the *Excel* button or by choosing the command options File, Close, Return to *Excel*. Our revised macro follows.

```
For I = 1 To 200
   Sum = 0
```

```
    For K = 1 To 100000
      Sum = Sum + K
    Next K
    Range("J3").Select
    Selection.Copy
    Range("I3").Select
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
      :=False, Transpose:=False
    Range("K3").Select
    Application.CutCopyMode = False
  Next I
```

We now want to link this macro to a button. To create the button, we select the View, Toolbars command and choose the option Forms. On the resulting toolbar we click on the button icon, drag out a button (Fig. 25), and in the resulting dialog box we assign the macro Move (Fig. 26).

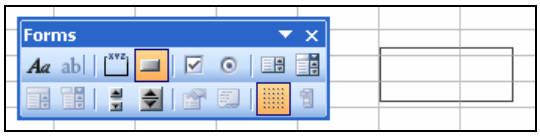


Figure 25: Creating a button

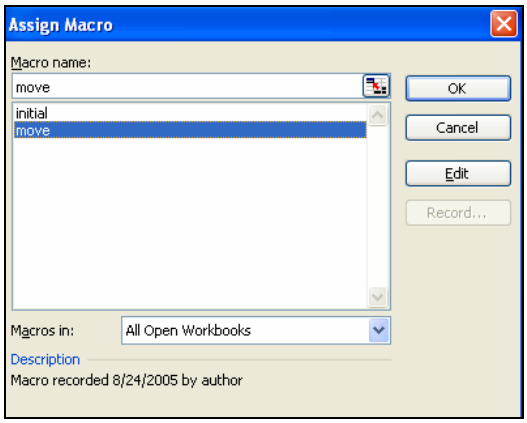


Figure 26: Assigning macro to button

Clicking on this button activates the macro, and we see the trace point and its tangent line move, with the slope of tangent constantly changing. This allows us to see how the slope of the line is related to the value of the derivative. It is also nice to incorporate a button to return to the initial condition. Consequently, we record another macro, Initial, where we start with Cell K3, click on Cell I3, enter 0, return to Cell K3, and turn off the macro recorder. We then link this to another button, marked Initial.

3 Polar and parametric graphs

We now use the animation techniques presented above to create eye-catching graphs of polar and parametric equations. This model not only allows us to create curves in a class such as calculus, but to implement the process in a way that closely parallels the way we would do it by hand. In the process, we also use this example as an opportunity to integrate some ideas from vector calculus and linear algebra. Another presentation of these ideas appears in [7].

Although *Excel* does not provide the polar graph type directly, it is easy to create these graphs as *xy*-charts by using elementary trigonometry. In the display of Figure 27 we provide a counter k, which gives degrees, in steps of size dk (usually 1) down the first column. In the second column, we use the *Excel* radians function to create the radian equivalent. We then enter a formula for r in the third column. Here, we create the polar

curve $r = a + b\cos(ct)$, where *a*, *b*, and *c* are entered as parameters. For this example we use $a = 0.5$, $b = 2$, and $c = 3$. Finally, in the next two columns we create the *x*- and *y*-values of points as $x = r\cos t$, $y = r\sin t$.

	A	B	C	D	E	F	G	H	I	J
1	r = a+bcos(ct)						trace point			
2	dk (step)	a	b	c			N	3		
3	1	0.5	2	3			x	y		
4							2.472	0.130		
5										
6				entire curve			development		trace	
7	k (deg)	t (rad)	r	x	y		X	Y	xx	yy
8	0	0.000	2.500	2.500	0.000		2.500	0.000	0.000	0.000
9	1	0.017	2.497	2.497	0.044		2.497	0.044	2.472	0.130
10	2	0.035	2.489	2.488	0.087		2.488	0.087		
11	3	0.052	2.475	2.472	0.130		2.472	0.130		
12	4	0.070	2.456	2.450	0.171		2.472	0.130		

Figure 27: Layout of polar graph construction

3.1 Arrow formula diagrams and graphing

Figures 28 and 29 provide descriptions of the formulas of the polar curve construction model.

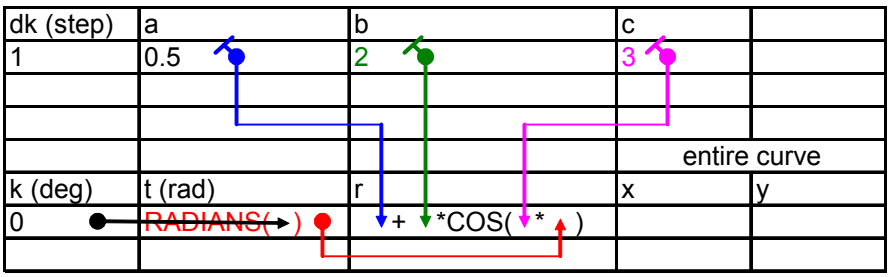


Figure 28: Polar equation formulas for *t* and *r*

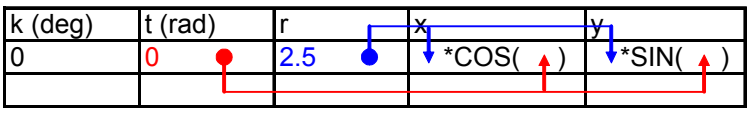


Figure 29: Polar equation formulas for *x* and *y*

We use the Chart Wizard to create the graph from Columns D:E shown in Figure 30. This process produces a nice polar graph, while the spreadsheet format provides us with a convenient medium through which we can experiment with values of the parameters.

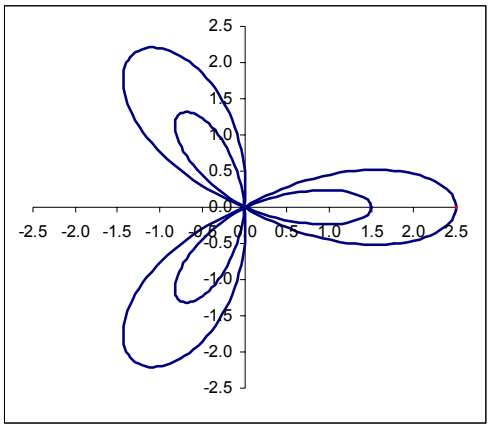


Figure 30: Graph of polar equation

3.2 Animating the model

However, the resulting curve appears instantaneously in a way that does not reveal the manner in which the equations generate it. Consequently, we will devise a way to use *Excel* to do this. As before, we enter a value of a point counter N. We now create a point-by-point development of the curve by generating only the first N points of the curve. Then, as we vary N, we can see the curve traced out one additional point at a time. One means of generating only the first N points of the curve is to create a second series of the x and y values whose formulas reproduce the values of x and y only for those points corresponding to $n \le N$. Otherwise, they simply repeat the values of the previous cell above. We present the formulas for the development curve in Figure 31.

dk	a	b	c			N	3
1	0.5	2	3			x	y
						2.472	0.130
			entire curve			development	
k	t	r	x	y		X	Y
0	0.000	2.500	2.500	0.000		IF(<= , ,)	IF(<= , ,)
1	0.017	2.497	2.497	0.044			
2	0.035	2.489	2.488	0.087			
3	0.052	2.475	2.472	0.130			

Figure 31: Development curve formulas

Once the development of Figure 27 is completed (see Fig. 34), we use the Chart Wizard to create the graph of the polar curve (D:E, blue), and then highlight and drag the new development series (G:H, red) into the existing chart. We also use the OFFSET function to generate the (x, y) coordinates of the trace point corresponding to N. We use these and the origin to produce within the graph a ray from the origin as a third series (I:J, green) that illustrates the angle, t, and the distance, r, of the trace point. Finally, we use either a slider or a macro approach to continually increase N and thereby to trace out the curve. We will see the new curve traced on top of the other (Fig. 32). However, for an even more effective display, we can right-click on the original curve and choose to show

neither marker nor lines, so that we see only the development curve as it is traced sequentially, as illustrated in Figure 33.

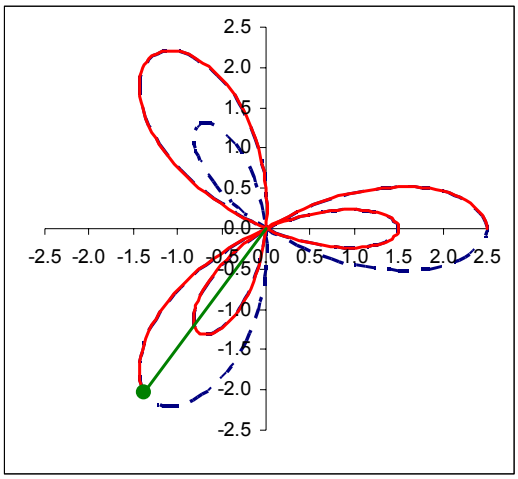


Figure 32: Showing both curves

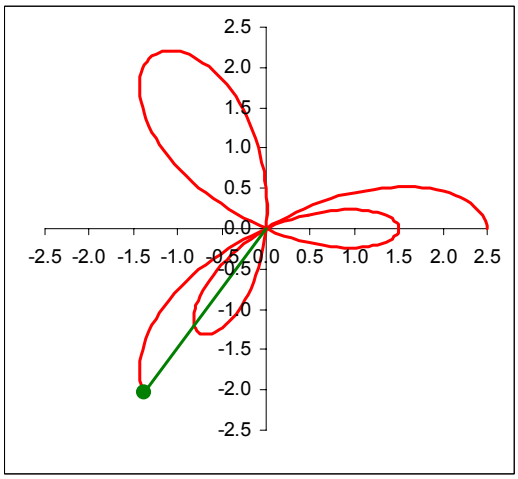
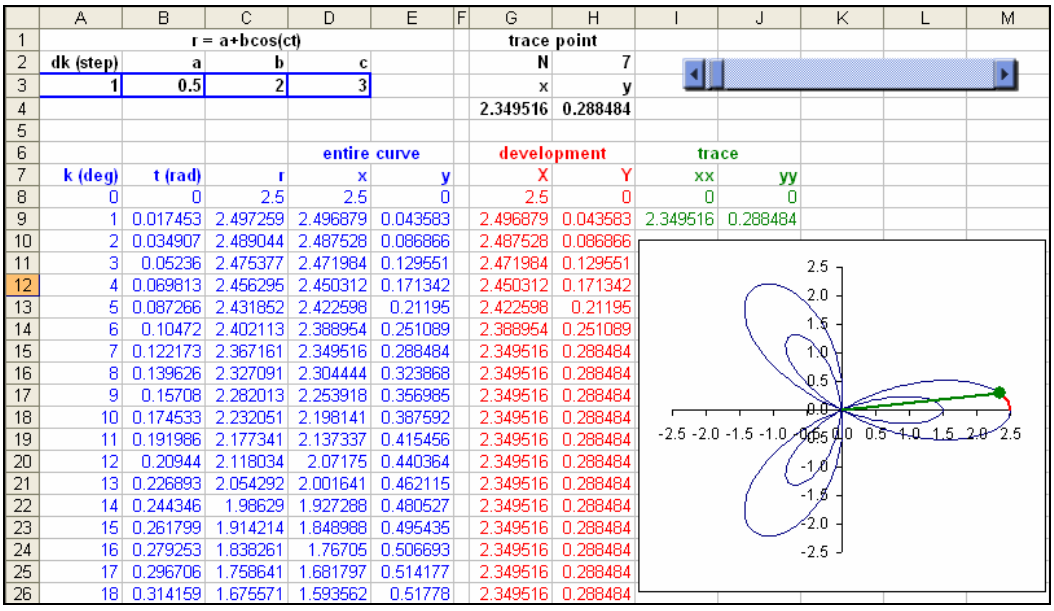


Figure 33: Tracing development curve

Figure 34 provides us with a screen shot the general layout for our model.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1			r = a+bcos(ct)				trace point						
2	dk (step)	a	b	c			N	7					
3	1	0.5	2	3			x	y					
4							2.349516	0.288484					
5													
6				entire curve			development		trace				
7	k (deg)	t (rad)	r	x	y		X	Y	xx	yy			
8	0	0	2.5	2.5	0		2.5	0	0	0			
9	1	0.017453	2.497259	2.496879	0.043583		2.496879	0.043583	2.349516	0.288484			
10	2	0.034907	2.489044	2.487528	0.086866		2.487528	0.086866					
11	3	0.05236	2.475377	2.471984	0.129551		2.471984	0.129551					
12	4	0.069813	2.456295	2.450312	0.171342		2.450312	0.171342					
13	5	0.087266	2.431852	2.422598	0.21195		2.422598	0.21195					
14	6	0.10472	2.402113	2.388954	0.251089		2.388954	0.251089					
15	7	0.122173	2.367161	2.349516	0.288484		2.349516	0.288484					
16	8	0.139626	2.327091	2.304444	0.323868		2.349516	0.288484					
17	9	0.15708	2.282013	2.253918	0.356985		2.349516	0.288484					
18	10	0.174533	2.232051	2.198141	0.387592		2.349516	0.288484					
19	11	0.191986	2.177341	2.137337	0.415456		2.349516	0.288484					
20	12	0.20944	2.118034	2.07175	0.440364		2.349516	0.288484					
21	13	0.226893	2.054292	2.001641	0.462115		2.349516	0.288484					
22	14	0.244346	1.98629	1.927288	0.480527		2.349516	0.288484					
23	15	0.261799	1.914214	1.848988	0.495435		2.349516	0.288484					
24	16	0.279253	1.838261	1.76705	0.506693		2.349516	0.288484					
25	17	0.296706	1.758641	1.681797	0.514177		2.349516	0.288484					
26	18	0.314159	1.675571	1.593562	0.51778		2.349516	0.288484					

Figure 34: Tracing a polar curve

At this point, we need to observe that if the parameter c is an integer, then with the value of dk set to 1 we will generate the entire curve of this model as k ranges from 0 to 360 degrees. However, if c is not an integer, then we may obtain only part of the curve. For example, in this model, if c is 2.5, then k must reach 720 degrees before we obtain the entire curve. In this case, to generate the entire curve we must either extend our model or set the value of dk to 2, so that the model plots points in increments of 2 degrees. In either case, we also must adjust the animation technique. In this example, in the developmental curve formulas of Columns G:H, we can replace the IF formula component: $k \leq N$ with: $k \leq dk*N$.

3.3 Adding creative graphical embellishments

Even though we have now produced a very nice animated visualization, in fact there is much more that we can do to embellish our diagram, to make it more interesting, and to exhibit additional mathematical information. To illustrate this, we will use Lissajous curves of the form $x = \cos mt$, $y = \sin nt$, where the parameters m and n are positive integers. For such parametric equations, we do not need to compute r. Instead, we enter formulas for x and y directly. In addition to our earlier computations, here we compute the direction angle of the tangent vector by finding derivatives of the equations for x and y with *Excel*'s arctangent function, ATAN2. Figure 35 provides us with a partial screen display of the layout.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1		x = cos(mt), y = sin(nt)									trace point			
2	dk	m	n							N	3			
3	1	3	5							x	y	angle	y scale	
4										0.988	0.259	1.668	0.1	
5														
6				entire curve						development		trace fly		
7	k	t(rad)	r	x	y	x'	y'	angle		X	Y	xx	yy	
8	0	0.000		1.000	0.000	0.000	5.000	1.571		1.000	0.000	0.978	0.358	
9	1	0.017		0.999	0.087	-0.157	4.981	1.602		0.999	0.087	0.976	0.354	
10	2	0.035		0.995	0.174	-0.314	4.924	1.634		0.995	0.174	0.970	0.342	
11	3	0.052		0.988	0.259	-0.469	4.830	1.668		0.988	0.259	0.961	0.323	
12	4	0.070		0.978	0.342	-0.624	4.698	1.703		0.988	0.259	0.950	0.299	

Figure 35: Animated fly and Lissajous model layout

We next create the image of a fly that we will use to trace out the image as it moves, and in the process, will indicate the direction of the tangent to the curve. To do this, in a convenient part of the spreadsheet we generate the image of a fly via the parametric equations $x = \cos 8t$, $y = \sin 4t \sin 5t$. To reduce the number of re-computations, and thereby increase the smoothness of the animation, we cut the number of points of the fly in half by using steps of 2 degrees.

To rotate this image through a variable angle, t, we can employ linear algebra (see [4]) to do this via a rotation matrix (1):

$$\begin{bmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{bmatrix} \qquad (1)$$

Because we arrange our vectors in rows, we multiply the fly's vectors on the right by this matrix in order to rotate its image. *Excel* supplies us with matrix operations that are automatically recalculated like the usual (scalar) *Excel* formulas, but we must learn a special technique for them. To multiply the position vector by the matrix, we first use the mouse to select the block of cells into which the resulting product vector goes. Then (Fig. 36) we type in =MMULT(and use the mouse to select the input vector. We next enter a comma as a separator, highlight the 2×2 rotation matrix, and press the F4 key to make that matrix an absolute reference (Fig. 37). We complete the formula by typing the right parenthesis,). However, because we are entering a matrix function, in *Excel* we finally must press the key combination Ctrl-Shift-Enter to enter, rather than just press the Enter key.

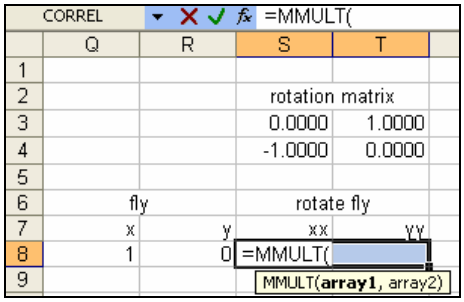


Figure 36: Matrix multiplication: I

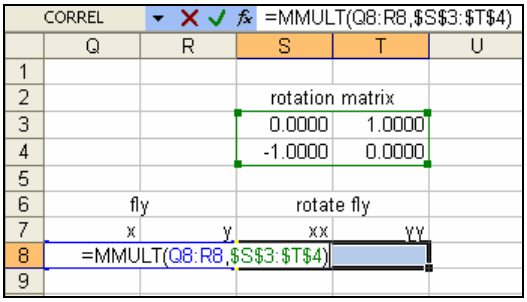


Figure 37: Matrix multiplication II

To complete the model we now use the development columns and the trace point. However, rather than simply plotting the trace point, we plot a translated, scaled image of the rotated fly by adding the scaled coordinates of the trace point to it. We show the creation of the x value of the trace fly below. The formula for y is similar. To achieve this, if x and y are the coordinates of the trace point and s is the scale, then we generate the corresponding coordinates of the nth point of the moving fly by $x + sx_n$ and $y + sy_n$. We employ this construction in Figure 38.

N	10								rotation matrix	
x	y	angle	fly scale						-0.4229	0.9062
0.866	0.766	2.007	0.100						-0.9062	-0.4229
development		trace fly					fly		rotate fly	
X	Y	xx	yy		k	t	x	y	xx	yy
1	0	+	*		0	0	1	0	-0.423	0.906

Figure 38: Moving fly image formulas

As usual, we now link the cell for the trace point N to a slider or incorporate it into an update macro. We show the original image of the fly in Figure 39, while in Figure 40 we see a stage in the animation process. We note that the fly points in the direction of the tangent.

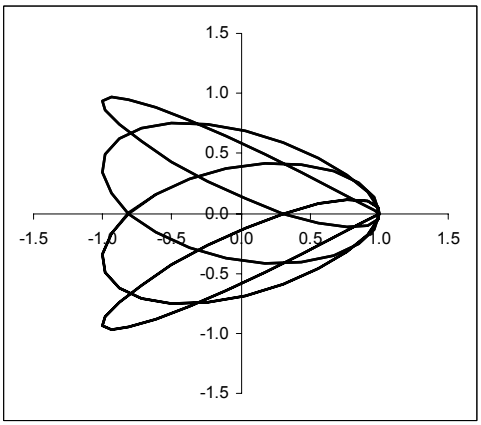


Figure 39: Base fly image

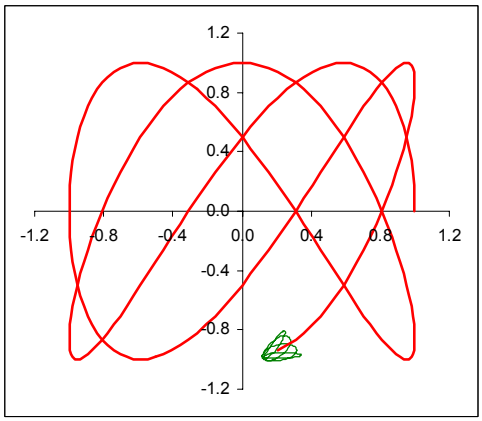


Figure 40: Animated output

4 Rectilinear motion

Our first example provided us with an animated model for investigating derivatives of a function in a traditional way. However, when we apply the ideas to such concepts as rectilinear motion, velocity, and acceleration, some students become confused in the interpretations of the various derivatives [5], [6]. Thus, we will combine the techniques of the first two models to create additional visualization images for our study. This model illustrates an approach that mathematics instructors can employ in designing models for classroom demonstrations to promote discussion while illustrating difficult or confusing aspects of a concept.

We assume that an automobile moves along a horizontal line so that a certain function, $y = f(t)$, gives its distance from a fixed point at time *t*. In our example, we use a function that will produce both positive and negative values of distance, velocity, and acceleration. We also could use other, simpler, functions to illustrate a smaller number of aspects. The function that we use is

$$f(t) = 10\exp(-0.2(t-5)^2) - 6\exp(-0.2(t-7)^2) + 2 \qquad (2)$$

We build our new model based on the techniques of first example.

Figure 41 provides us with an overview of the main part of the layout. Columns A:E provide the points for the location function and its first two derivatives; Columns P:Q for the image of an automobile; and I:J for two scaled and translated images of the automobile. We will discuss the use of the automobile later on.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1																	
2	Enter function and derivatives								Point Counter							Scale car image	
3										current	next						
4									N	100	101					scale 1	0.3
5										Current Values				INITIAL		scale 2	0.15
6									t	y	y'	y''					
7									2.00	3.61	1.90	1.57		MOVE		label	v = 1.90
8	dx	0.02															
9																	
10	Curve graph and derivatives								Car Display 1		Tires Display 1		Trace Point			base car	
11	n	t	y	y'	y''				x	y	x	y	x	y		x	y
12	0	0.00	2.067	0.134	0.240				0.013	0	2.713	0	2	3.613		-12	0
13	1	0.02	2.070	0.139	0.248				0.613	0	0.913	0	2	0		-10	0
14	2	0.04	2.073	0.144	0.255								0	3.613			
15	3	0.06	2.076	0.149	0.263				1.213	0						-8	0
16	4	0.08	2.079	0.154	0.271				2.413	0						-4	0
17	5	0.10	2.082	0.160	0.279												
18	6	0.12	2.085	0.165	0.288				3.013	0						-2	0
19	7	0.14	2.088	0.171	0.297				3.313	0						-1	0
20	8	0.16	2.092	0.177	0.305				3.613	0.6						0	2
21	9	0.18	2.095	0.184	0.314				0.313	0.6						-11	2
22	10	0.20	2.099	0.190	0.324				0.013	0						-12	0
23	11	0.22	2.103	0.196	0.333												
24	12	0.24	2.107	0.203	0.343				0.913	0.6						-9	2
25	13	0.26	2.111	0.210	0.353				1.063	1.05						-8.5	3.5
26	14	0.28	2.115	0.217	0.363				2.413	1.05						-4	3.5
27	15	0.30	2.120	0.225	0.373				2.713	0.6						-3	2

Figure 41: Rectilinear motion model layout

4.1 Designing imaginative graphic displays

Often in texts, we will see the concept of rectilinear motion discussed through the traditional graphs of Section 2. Now, however, we will create a traditional speedometer image to accompany those graphs to help us to interpret the effects of changes in velocity. Initially the trace point of the car moves to the right, so the dial of the speedometer moves into the right, indicating a positive velocity. As time increases, we see the dial moving further to the right, which indicates an increase in velocity (*i.e.* positive acceleration). Later, while the speedometer dial remains positive, indicating that the car is still moving in a positive direction, we will see that its speed is slowing. This corresponds to a negative acceleration. The point at which the automobile's speed ceases to increase corresponds to an inflection point on the original curve, a relative maximum on the velocity curve, and a zero of the acceleration curve. We continue our investigation in a similar fashion and observe when the car is moving backwards.

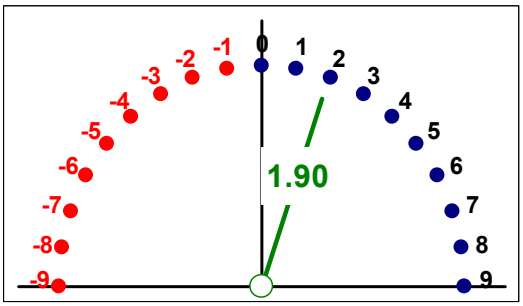


Figure 42: Animated speedometer drawing

To create the speedometer background, we use polar coordinates to plot points in a semicircle (Figs. 43-44). To get two different colors, we can either plot two separate series, or recolor half of the points individually. So that the speed labels appear in an esthetically pleasing format, we have generated a second, slightly larger semicircle on which to attach labels, and then hide the graph's markers and lines. The only somewhat complicated construction is that of the speedometer dial. We leave that as a challenge to readers, although its formula also appears in the associated *Excel* model.

Speedometer (dots)			
radius	5		
deg	rad	x	y
0	RADIANS( )	*COS( )	*SIN( )
10+			

Figure 43: Speedometer (formulas)

Speedometer (dots)			
radius	5		
deg	rad	x	y
0	0.00	5.00	0.00
10	0.17	4.92	0.87
20	0.35	4.70	1.71

Figure 44: Output

As a second visualization device, we show the progress of a car along a horizontal line (Fig. 45). Here we can see that the car moving to right corresponds to increased y-values on the original graph. To show this we create a simple picture of a car, and then generate a scaled copy of it at a trace point that move along the x-axis. We use the same technique that we used with the fly earlier, except that to simplify the image we use circle graph markers for the tires. In addition, we also provide another image of the automobile moving with a trace point along the y-axis of the original curve (Fig. 46).

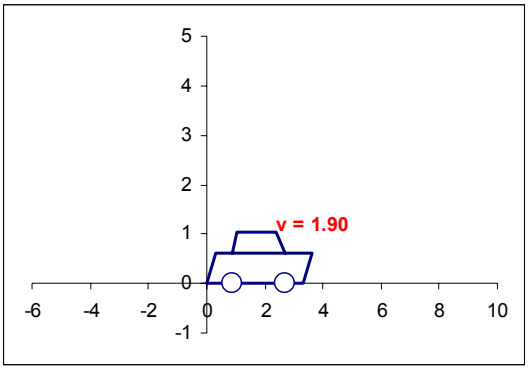


Figure 45: Image showing position

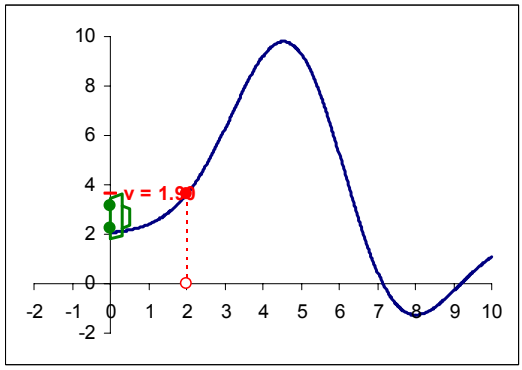


Figure 46: Graph of position vs. time

In a standard analysis using only the function graphs, some confusion can arise because the distance traveled in the original graph appears on the *y*-axis, since that horizontal axis represents time. Consequently, in Figure 46 we include the image of the automobile along the *y*-axis of the original curve as well as the automobile moving by itself, to show the connection between the two visual representations.

5 Graphs visualizing a calculus proof

We can create animated *Excel* visualizations related to mathematical proofs as well as for applications. As an illustrative example, let us consider the Mean Value Theorem. We use this example to show students one method for discovering a proof. We begin by creating the graph of the cubic function, $f(x) = x^3 - 2.4x^2 + 1.44x + 0.5$, $0 \le x \le 2$. We also plot the line that connects the endpoints. As in our first example, we create a tangent line segment at a trace point linked to a slider. We display the values of the slope on each of the lines. As we move the slider, we can see the places where slopes are approximately equal. Since we do this using the point counter, *N*, to scroll through points, we may not be able to find the exact location through scroll bar. We could also use the spreadsheet's solver tool to overcome this difficulty in locating the points, although we will not pursue that here (see [7]). Figure 47 gives us an overview of the screen display.

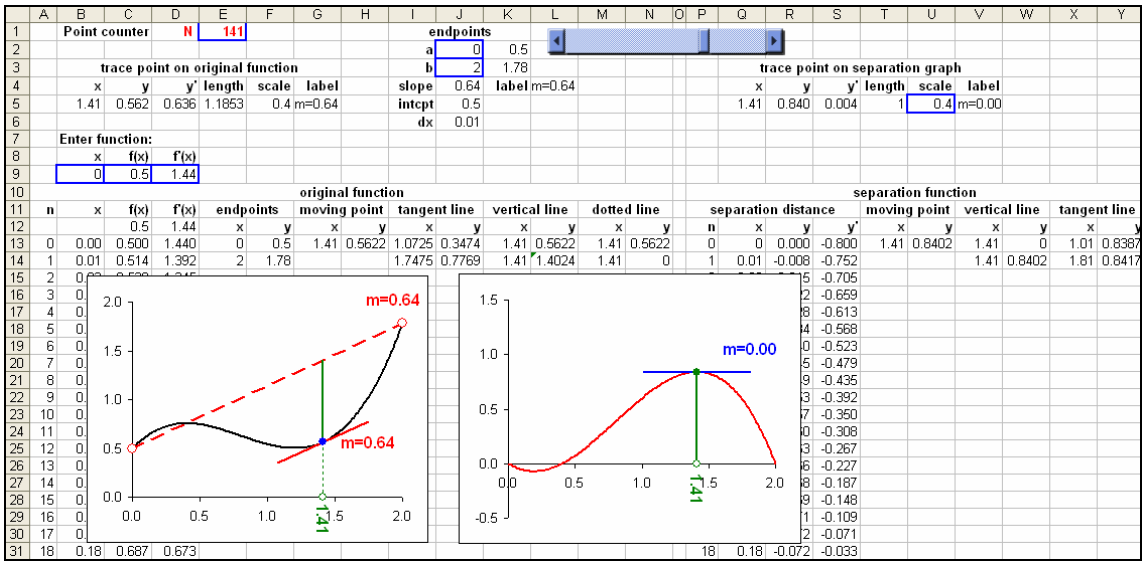


Figure 47: Mean value theorem model

5.1 Data tables

In this example, we introduce another powerful *Excel* tool, the Data Table, to construct our function. This will allow us to change functions more easily by simply changing the function and its derivative once without needing to copy the changes. We first use three defining cells to contain a generic value for x and the corresponding computed values of $f(x)$ and $f'(x)$, as shown in Figure 48.

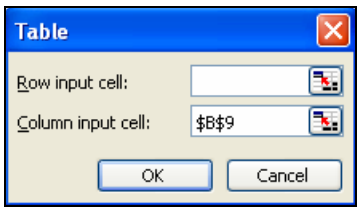
Enter function:		
x	f(x)	f'(x)
0	^3-2.4* ^2+1.44* + 0.5	3* ^2-4.8* + 1.44

Figure 48: Defining cells for function and derivative

We next create a table of values for x, $f(x)$, and $f'(x)$. We initially leave the top row of the table empty (Fig. 49). We then generate a column of x values just as before. However, into the two right entries of the top row of the table we enter formulas that reproduce the value of the function and its derivative. We then use the mouse to select the three columns (Fig. 50) and choose the command Data, Table. In the ensuing dialog box of Figure 51, we enter the cell of the generic x-value (here B9) as the Column Input Cell. When we press the OK button, *Excel*'s Data Table command repeatedly substitutes the successive values of x into the defining formulas for $f(x)$ and $f'(x)$, and returns the results into the corresponding cells of the table's last two columns. To vary the example for use with other functions, we need only change the formulas for $f(x)$ and $f'(x)$ in the two defining cells.

	Enter function:		
	x	f(x)	f'(x)
	0	0.5	1.44
original function			
n	x	f(x)	f'(x)
0	0.00		
1	0.01		
2	0.02		

7		Enter function:		
8		x	f(x)	f'(x)
9		0	0.5	1.44
10				
11	n	x	f(x)	f'(x)
12			0.5	1.44
13	0	0.00		
14	1	0.01		
15	2	0.02		
16	3	0.03		
17	4	0.04		



Figures 49-51: Data table construction

5.2 Discovering a proof

As we move the slider, we may observe that the places where the derivative matches the slope of the line seem to correspond to places where the line and the curve are the furthest apart. To examine this, we extend our drawing by showing the vertical line between the line and the curve at a point x, and extend it via a dashed line to the x-axis (Fig. 52). We next create a second graph (Fig. 53) that shows the vertical distance be-

tween the line and the curve at each point, together with a trace point and tangent line. As we move slider, it appears in fact that the point at which the slopes of the lines in the first graph match occurs at a local maximum of the distance function. From this observation, we can look at the equation of the latter function and use it to develop a formal mathematical proof of the Mean Value Theorem (see [10], p. 257). Thus, our approach can help in discovering a proof and in seeing its underlying geometry.

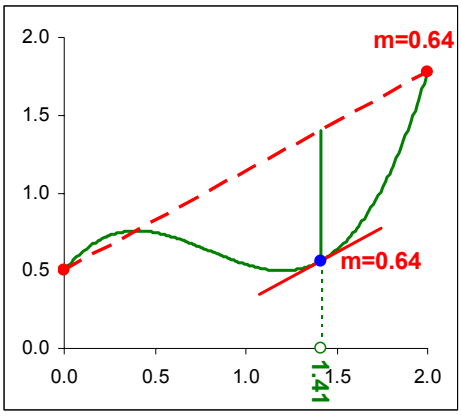


Figure 52: Mean value theorem

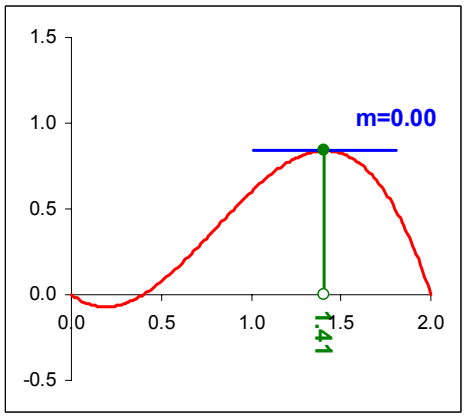


Figure 53: Distance function

We create the various components of this model using the techniques of our earlier examples. There are some differences, however. In this example the endpoints a and b are parameters of the model. Since we plot points numbered from 0 to 200, we compute the increment between x-values as $dx = (b-a)/200$, the slope of the connecting line as $m = (f(b) - f(a))/(b-a)$, and its y-intercept as $y_0 = f(a) - ma$.

6 Other issues

Ultimately, we will probably want to polish the format of our models. The *Excel* file Example5.xls provides us with descriptions of some formatting techniques to use in enhancing graphic displays.

Experienced spreadsheet users will observe that we have not used names to identify cell references in our formulas, although many people prefer to do that, thereby embracing standard programming conventions. However, as discussed in [7], we prefer to describe the spreadsheet construction process in a way allows students to encounter mathematics without needing to rely upon an algebraic notation. While algebraic notation is natural for mathematicians, it is not necessarily innate for many students. Thus, in general, we try to reverse the traditional algebra-first approach and instead use the spreadsheet creation process to introduce algebraic concepts.

Excel provides a radar graph type that in some ways is similar to a polar graph. However, rather than plotting points by their (r,θ) coordinates, it plots points at regularly spaced angles (of size $360/n$ degrees with n points) around a central point. Unfortunately, the generation of points starts at the top of a circle, rather than at the usual location for $\theta = 0$. If we set the minimum radar graph scale value for r to 0 (otherwise the center is set to the minimum value of r so that even negative values are plotted outwardly from the center producing incorrect curves), we can plot those polar curves that can be formed in one pass around the origin. However, we cannot use a radar graph in a

simple manner to construct such curves as $r = \cos(2.5t)$ (which require two passes around the origin), curves given by parametric equations, or curves not plotted at constant angle frequencies. In addition, adding such useful embellishments as tangent and normal lines are not straightforward processes, and animation effects are more difficult to implement using this graph type.

7 Concluding remarks

This paper provides demonstrations of some of the fundamental techniques to employ in *Excel* to create animated graphical displays for teaching mathematics. Of course, before we employ a new approach we should first examine the reasons for doing so. In particular, it would seem that a computer program, such as *Excel*, should meet at least three criteria. First, it should fit the mathematics that we teach and enhance the learning process. Second, teachers and students should need to devote only a limited amount of time to learning to use the software effectively. Third, the software should be usable in later courses and throughout a student's career. A spreadsheet, such as *Excel*, meets these criteria quite well.

First, we typically carry out many standard mathematical algorithms and computational concepts in a table format like that of a spreadsheet. Moreover, we can implement numerous mathematical techniques, including iterative and recursive algorithms, directly in *Excel*. More than this, we can employ the actual spreadsheet creation process to introduce concepts from algebra, calculus, and differential equations, during the designing process [7].

Further, graphic displays are invaluable for learning and communicating mathematics. Most of us use pictures to conceptualize ideas such as the mean value theorem, area, eigenvectors, rates of change, definite integrals, and solutions of initial value problems. Spreadsheets provide us with high-quality graphics that allow us to increase the effectiveness in conveying ideas by moving from examining static displays to interacting with animated graphs while investigating such topics as convergence, changes in parameters, and the predictions of models for population growth, the spread of epidemics, and predator-prey interactions [7]. (For a look at an earlier approach to animation using circular references, see [1].)

Regarding the second criterion, most students at the tertiary level will already be familiar with the basic usage of *Excel*, and can quickly learn the additional techniques used in designing effective graphics. The time required for teachers to become competent designers is reasonable, too. Nonetheless, those who create original animation models should be aware that the time required to design a truly sparkling presentation will be much greater than that needed to draw a quick sketch on the board. Thus, for classroom demonstrations, it is important to use the approach wisely, and to build a library of successful models for future use. Fortunately, many of these are already available [1], [3], [9]. As an added benefit for educators, designing spreadsheet illustrations provides an avenue for invigorating professional growth, and an opportunity to develop and present new ideas.

Third, because learning spreadsheet operation need not impose a significant burden on class time, especially in using teacher-designed classroom demonstrations, we can incorporate spreadsheets into a wide range of mathematics classes. Having teachers in subsequent classes continue to employ it does require consultation, but most students certainly will encounter spreadsheets in their future employment, where spreadsheet

skills are valued highly. We realize that other mathematical software is also very effective for teaching mathematics. However, relying solely on dedicated mathematics software may have unexpected disadvantages. Often the workplace does not support specialized mathematics software, and using specialized mathematics software can reinforce the unfortunate view that doing mathematics is "different" from everyday life, and needs its own special tools.

Finally, the author has encountered at least two additional cultural benefits from the use of *Excel* in teaching mathematics. First, it has proven to be exceptionally appropriate tool for teaching in a developing nation, where spreadsheets are the only mathematical tool readily accessible on computers, and students know that they need spreadsheet skills to find future employment. Second, it has made it easier for some students to discuss mathematics with their parents via a tool that is familiar to both of them, allowing the students to demonstrate a variety of new and useful techniques, such as animation, data tables, and matrix operation.

8 Projects

Many of the references below direct us to other sources of good examples (see especially [7] and [9]). Here we list some that are closely related to the ideas listed in the paper.

Curve Sketching: Create an animated graphing model that incorporates, and improves upon, a variety of the features typically found on graphing calculators. You may wish to include *Excel*'s solver and goal seek commands in your models. As a second project, create an animated graphic showing the osculating circles of a function used in the study of curvature.

Parametric Equations: Create the image of a pointing finger to trace out a curve. A series of TV promotions on the Australia Broadcasting Corporation during the late 1990's inspired the animated drawing in Figure 54. In addition, create animated graphs of epicycloids and related curves (see [1], [2], [3]).

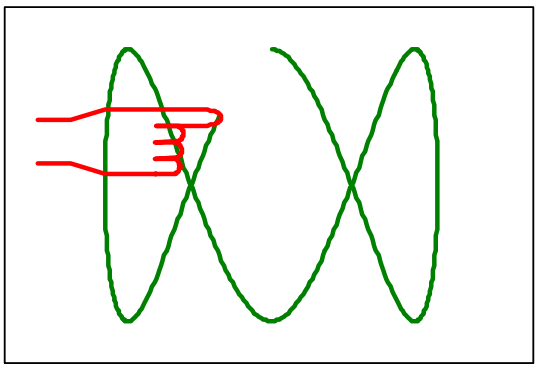


Figure 54: Animated ABC logo construction

Moving Images: Create an animated spreadsheet visualization of a pursuit problem that includes airplanes, missiles, or similar flying objects.

Visualizing Proofs: Create an animated model to illustrate the Fundamental Theorem of Calculus or another mathematical theorem from calculus, linear algebra, or geometry.

Algorithm Illustration: Design an animated visualization for an algorithm from computational mathematics. A few of the many possible topics include numerical integration, Newton's method, Euler's method, the Gauss-Seidel algorithm, and the 2-dimensional simplex algorithm.

References

[1] Arganbright, Deane (1993). *Practical Handbook of Spreadsheet Curves and Geometric Constructions*. CRC Press, Boca Raton, FL, USA, 1993. ISBN: 0-8493-8938-0.

[2] Gray, S., et al. *National Curve Bank*: http://curvebank.calstatela.edu/home/home.htm

[3] Hill, David and Roberts, Lila. *Demos with Positive Impact*: http://mathdemos.gcsu.edu/

[4] Lay, David C. (2003). *Linear Algebra and its Applications, 3rd. ed*. Addison-Wesley, Boston. ISBN: 0-201-70970-8.

[5] Monk, Steven (1992). Students' Understanding of a Function Given by a Physical Model. In G. Harel & E. Dubinsky (eds), *The Concept of Functions: Aspects of Epistemology and Pedagogy*, 175-194), MAA Notes 25. Washington, DC: Mathematical Association of America.

[6] Monk Steven and R. Nemirovsky (1994). The Case of Dan: Student Construction of a Functional Situation Through Visual Attributes. In E. Dubinsky et al (eds), *Research in Collegiate Education – I. CBMS Issues in Mathematics Education* 139-168. Providence, RI. American Mathematical Society.

[7] Neuwirth, Erich and Deane Arganbright (2004). *The Active Modeler: Mathematical Modeling with Microsoft Excel*. Thompson/Brooks-Cole Publishers, Belmont, CA, USA. ISBN 0-534-42085-0.

[8] Neuwirth, Erich (1995). "Spreadsheet structures as a model for proving combinatorial identities". *Journal of Computers in Mathematics and Science Education*, 14(3), 419-434.

[9] Neuwirth, Erich. *Mathematical and Educational Applications of Spreadsheets*: http://sunsite.univie.ac.at/spreadsite/

[10] Weir, M., Hass, J., and Giordano, F. (2005), *Thomas' Calculus, 11th ed*. Addison-Wesley, Boston. ISBN 0-321-24335-8.

[11] Math DL, The MAA Mathematical Sciences Digital Library, Mathematical Association of America: http://www.mathdl.org/jsp/index.jsp