

Unit VI

Derivatives and Differential Equations

These materials were developed for use at Grinnell College and neither Grinnell College nor the author, Mark Schneider, assume any responsibility for their suitability or completeness for use elsewhere nor liability for any damages or injury that may result from their use elsewhere.

Unit VI

Derivatives and Differential Equations

The language used to set up virtually all of the problems in physics is the language of differential equations. Mechanics is based on $F=ma$, a differential equation. Electricity and Magnetism is based on Maxwell's equations, a set of four vector differential equations. Quantum Mechanics is based on the Schrödinger equation, a complex differential equation. In undergraduate theory courses, we spend virtually all of our time on the few equations and examples that can be solved analytically, which constitute a tiny fraction of the real problems the world presents us with. The insight we gain from the complete analytical solution is formidable, but there are some systems in which we simply cannot approximate the real equations with harmonic oscillators or constant forces, so numerical solutions become essential. Before we study the methods of solution of differential equations, first let's look at the calculation of derivatives numerically. In this realm we can more easily understand some simple improvements to the most obvious algorithms, and then we can invert those algorithms to solve differential equations.

Guidebook Entry VI.1: Numerical Derivatives

The derivative is defined as the limit

$$dy/dx = \lim (\Delta y / \Delta x).$$

We can use this expression and eliminate the limit to gain an approximation to the derivative that is useful for real data. In other words

$$dy/dx \approx (\Delta y / \Delta x).$$

Let's use this and Excel to estimate some derivatives for the sine function. Create a spreadsheet that gives the first period for the sine function, with points spaced every .2, or about 30 points over the period.

Make another column that estimates the derivative by the ratio of the differences. In other words, if the x value is in A5, the $\sin(x)$ value in B5, we would estimate the derivative in the adjacent cell C5 to be

$$C5 = (B5-B4)/(A5-A4).$$

Create a full column like this. Then, in the next column, put the values of the analytical derivative of the sine function. How do the two compare? You may wish to create a column that gives the error in the numerical estimate, as we did in our interpolation exercises.

Graph the numerical and analytical derivatives on the same graph. What do you notice in your comparison? Explain any small discrepancies, and discuss your explanation with your instructor.

You should have noticed a phase shift of one-half cell between the numerical and analytical functions. How might you correct for this? See if you can now make the numerical and analytical functions agree more closely. Describe what you observe.

We can also make more reasonable approximation of the derivative at a point by taking the difference between the function values to either side. For example, the derivative at the x value give in cell A5 would be given by the expression

$$C5 = (B6-B4)/(A6-A4).$$

Calculate the derivative this way, and compare the accuracy to the previous method.

This latter technique suffers in accuracy some, but has the nice feature that the x values are preserved across the rows. What happens to the endpoint values in each case, that is, the derivatives at the very smallest and very largest x values?

You should have found a considerable improvement in the agreement between numerical and analytical expressions when you took a "half-step" derivative. This is in fact closely related to the concept called the intermediate value theorem which you learned about in calculus; the slope of a continues function matches the slope of the secant line somewhere between the two endpoints. Our best guess for the

location of that point, knowing nothing else, is the midpoint. Unfortunately, this becomes slightly more awkward to handle in a spreadsheet, since we don't then have nice x labels in our first column.

Guidebook Entry VI.2: Inverting a Derivative--Solving a First Order Equation

Let us consider the simple problem provided by viscous drag. In this case, we have a force law given by $F = -\gamma v$. Let's imagine that we only care to solve for the velocity as a function of time, and not the position. Given that, we can simply solve for the acceleration as dv/dt , and use Newton's second law to relate this to the force. Express this time derivative of v in terms of γ , m , and v .

If we have a starting value for v , we can estimate the value of v a tiny bit of time later through our approximate derivative formula. Do this, using $dv/dt \approx (v_i - v_{i-1})/\Delta t$.

Now we can convert this into an expression for a spreadsheet to update a velocity value. That is, if our fifth value of v is in cell B6, and the corresponding acceleration values in the C column, we can update the v value which should be placed in B7 as

$$B7 = B6 + C6*(A7-A6),$$

where I have assumed that the time values are in the A column. Use this, with the simplifying assumption that $\gamma/m = 1$ to create a first numerical integration spreadsheet. In this case, the acceleration values in column C are the same in magnitude as the corresponding velocity values, but opposite in sign. Start with an initial v value of 5. Fill the sheet down, and graph and describe your result.

Compare your result to the analytical solution e^{-t} .

For this differential equation, it is difficult to get convenient half values, but we can use our every-other-value method from the last guidebook entry profitably here. In other words, as soon as you can, use the formula

$$B_7 = B_5 + C_6*(A_7 - A_5).$$

How do the results compare with the other method for the first ten or so values? How well does it fare in the long term?

You should have had your first brush with a numerical instability. Some of these are inherent in the numerical methods (as is the case in that example), and some are inherent in the differential equation being solved (especially those that admit increasing exponential solutions).

Most of the differential equations we face in physics are second order differential equations, such as when we use Newton's second law to solve for the position as a function of time. In the next exercise, we will extend our techniques to these sorts of equations, and find that the second order nature, being a nuisance in some ways, turns out to be particularly well-suited to the use of the half-step algorithm.

Guidebook Entry VI.3: Second Order Equations

For a second order differential equation, we have an expression that relates the second derivative to the other parameters of the motion--position, velocity, time. To do this, we clearly need to keep track of each of these parameters. Let us look first at a differential equation we know well, the harmonic oscillator. This comes from Newton's second law and Hooke's law:

$$F = ma = -kx$$

which becomes

$$\frac{d^2 x}{dt^2} = -\frac{k}{m} x.$$

The familiar solutions to this are the sine and cosine functions, with an angular frequency of

$$= \sqrt{k/m}.$$

Just to make sure you all are comfortable with this (especially those of you who haven't taken a lot of physics yet), verify that this indeed satisfies the differential equation analytically. Write the most general solution as well.

Now, let's see how we might solve this numerically. To solve the problem numerically, we are actually solving for a specific solution, rather than a general solution as you gave above. Use your result above to write the specific solution that has $x_0 = 5$ and $v_0 = 0$.

Now, create a spreadsheet that has columns labeled for t , x , v , and $a = F/m$. For the latter, you will need to have an explicit value for k/m ; for convenience, choose $k/m = 1$. Make a cell that contains the time spacing, which is then used consistently throughout the sheet, so you can vary that time interval. Use our simplest algorithm from above, often referred to as Euler's method, applied to the velocity:

$$v_1 = v_0 + a_0 \cdot t$$

and applied to the position

$$x_1 = x_o + v_o t.$$

Choose the time spacing such that there are at least 20 points in a full period. Look at at least three periods worth of the resulting function. Describe what you see. In what ways is the approximation good? In what ways is it bad?

Put in a column that has the analytical solution, and another column that shows the difference between the numerical and analytical solutions. Describe what happens to the error when you cut your step size in half. Make sure to compare the accuracy at the same time values.

We can do a much better job by applying a half-step version of the Euler method. In this case, we have, for at least later steps of Euler (I'll choose the form for updating the fifth point):

$$v_{4\frac{1}{2}} = v_{3\frac{1}{2}} + a_4 t$$

and

$$x_5 = x_4 + v_{4\frac{1}{2}} t.$$

This has the somewhat unpleasant effect of making the v values apply to the half integral values of the time. It also requires some conventional Euler starter steps. Try to work your way through this, and check your work with your instructor. Does this seem to work better than the simple Euler method?

Again, see how much your approximation improves when you cut the step size. In relative terms, is the improvement better or about the same for the half-step Euler?

Guidebook Entry IV.4: How Close?--Estimating Errors

As we have seen before, Taylor's series is a useful way to understand the accuracy of our approximations. We are trying to find the value of x_1 at a t value increased by Δt based on the value of x_0 and various derivatives. First, write the Taylor series through terms involving x_0''' (the third time derivative of $x(t)$ evaluated at $t = 0$).

The sequential terms of this series are presumed to get smaller and smaller by virtue of the higher powers of Δt , which we assume to be a small number. So, the first term not included in an approximation is a reasonable estimate of the error, and the power of Δt tells us how quickly the approximation improves as we cut the size of Δt . Consider our simplest Euler approximation. How much of the Taylor series is included? What is the first error term?

Can you use this to explain your results when you cut the step size in half earlier?

Now we will look at the half step approximation. Here we used the derivative at the half step. We need to know what that derivative is in terms of the derivatives at the zero step, so justify the following first order approximation in terms of the Taylor's series expansion of the derivative function:

$$x_{\frac{1}{2}}' = x_o' + \frac{1}{2} x_o'' .$$

Plug this expression into your Taylor's series expression above, and identify the error term.

Can you make sense of the improvement you saw in the half-step approximation of the harmonic oscillator?

Finally, let us see how one goes about using Mathematica to get numerical solutions to the harmonic oscillator.

Guidebook Entry IV.4: Using Mathematica to Solve Differential Equations Numerically

Let's say we wish to solve the differential equation

$$x'' = -x$$

which we know as the harmonic oscillator equation we investigated above. Mathematica will solve this numerically given the command

```
NDSolve[{x''[t]==-x[t], x[0]==5, x'[0]==0},x,{t, 0, 13}]
```

This produces an "interpolating function" from which Mathematica can extract values. One can get a particular value, say at $t = 4$ from a just previously calculated function using the command

`x[4] / . %`

where the characters `/ .` say use the rules developed in the interpolating function, and `%` refers to the previous result. If one wants an earlier line, say line 5, one would instead use `%5`. To actually get a graph of the function, one uses the command

`Plot[Evaluate{x[t] / . %}, {t, 0, 7}]`

Do a numerical evaluation of the harmonic oscillator equation, and plot it over the first two periods.

The equation $f(x) = x - \tan(x)$ has a root at $x = 0$, which you can easily verify. How well does Mathematica find that root? Why is there difficulty here? Would you have any indication from Mathematica that there may have been a problem?