

# Math 2280 - Lecture 6

Dylan Zwick

Fall 2008

## 1 Euler's Method

For the first order differential equations we've seen so far, most of them have had the form:

$$\frac{dy}{dx} = f(x, y)$$

and most of them we've been able to solve using one technique or another to get the solutions. Now, differential equations for which we've got explicit solutions are actually the exception rather than the rule, and even for relatively simple looking differential equations it may be impossible to figure out the solution. For example:

$$\frac{dy}{dx} = e^{-x^2}$$

has no solution  $y = f(x)$  where  $f(x)$  is an elementary function. By elementary function we mean a function that can be expressed in terms of the standard functions (exponentials, cosines, logarithms, polynomials, etc...) from calculus.

However, even in situations where we cannot figure out the explicit solution, we can frequently construct approximations to these solutions. One of the oldest methods used to figure out approximate solutions is called "Euler's Method", named after the (extremely) prolific and influential mathematician Leonhard Euler.

## 1.1 Euler's Method Algorithm

The idea behind Euler's method is pretty simple. We're given an initial value problem:

$$\begin{aligned}\frac{dy}{dx} &= f(x, y) \\ y(x_0) &= y_0\end{aligned}$$

and we want to construct an approximate solution. We construct the approximate solution by starting with our initial condition, and taking the slope at that point  $f(x_0, y_0)$ . We assume that slope will be constant over a small change in  $x$ , and so we move forward a small distance  $h$  in the  $x$ -direction. The size of  $h$  is called the "step size" for our implementation.

If we move an amount  $h$  in the  $x$ -direction, and our slope is  $f(x_0, y_0)$ , then we move an amount  $h \times f(x_0, y_0)$  in the  $y$ -direction. This gives us an approximation for the value of our solution for the input value  $x_0 + h$ , namely  $y_0 + h \times f(x_0, y_0)$ . We call these approximated points  $(x_1, y_1)$ . Then from there, we just continue in the same fashion.

What we get is a sequence of line segments that approximate our solution curve. If our step size is very small, this sequence of line segments looks more and more like a curve, and (in theory) they get closer and closer to our actual solution curve.

## 1.2 Example

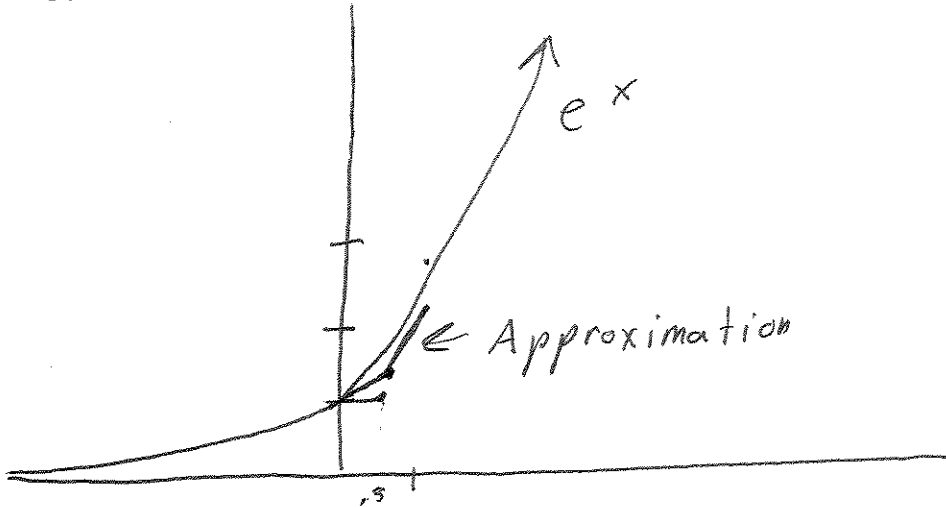
Let's say we start with the simple differential equation:

$$\begin{aligned}\frac{dy}{dx} &= y \\ y(0) &= 1.\end{aligned}$$

This initial value problem has solution  $y(x) = e^x$ . Let's check out what Euler's method predicts for the solution at  $x = 1$  using a step size of  $h = .5$ . Applying the Euler's method algorithm we get the following table:

$n$	$x_n$	$y_n$	$f(x_n, y_n)$	$e^{x_n}$
0	0	1	1	1
1	.5	1.5	1.5	1.65
2	1	2.25	2.25	2.72

So, we can see that Euler's method gives an O.K. solution here, but the approximation isn't great.



If we do this again, only instead use a larger step size of  $h = .1$  we have to do quite a few more calculations, but our estimate improves:

$n$	$x_n$	$y_n$	$f(x_n, y_n)$	$e^{x_n}$
0	0	1	1	1
1	.1	1.1	1.1	1.11
2	.2	1.21	1.21	1.22
3	.3	1.33	1.33	1.35
4	.4	1.46	1.46	1.49
5	.5	1.61	1.61	1.65
6	.6	1.77	1.77	1.82
7	.7	1.95	1.95	2.01
8	.8	2.14	2.14	2.23
9	.9	2.36	2.36	2.46
10	1	2.59	2.59	2.72

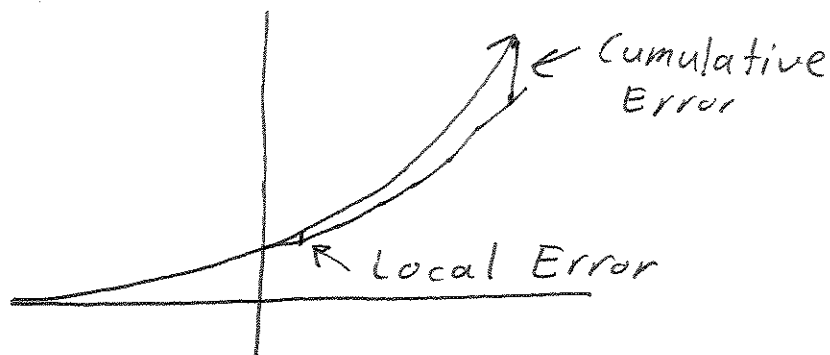
So, as we can see, by taking a smaller step size we get closer to the actual value, although again we're a little bit off. If we took an even small step size, say  $h = .01$  then we'd get even closer.

Now, this example has been with a very simple situation, for which we've already got the solution, but the idea is that it illustrates the method. For more complicated problems with more many more steps these computations can get very, very tedious and time consuming. One of the major reasons we have computers is so the computer can do this for us. You'll be doing some of this stuff in your first Maple project.

### 1.3 Sources of Error

There are two major sources of error in the use of Euler's method, called *local error* and *cumulative error*.

Local error is the result of our assumption that the slope is constant over our small step size  $h$ . Now, if our function  $f(x, y)$  is continuous and our step size is small, this isn't an unreasonable assumption, but it's not exact, and this will introduce some error.



The other error results from cumulative error. Because in each of our steps we introduce some local error, the starting points from which we calculate the slopes for each step are also not quite right, and so the slopes we calculate are not quite right, and so this introduces even more error. The overall cumulative effect of this error is called, not terribly creatively, cumulative error. It's meant to represent the total error (distance) of our approximate solution from the actual solution.

Now, frequently we don't know what the actual solution is, and so we just want to know that our approximation is within some range of values of the actual solution. The study of this type of situation along with related situations is a topic for an entire class on error analysis. A necessary class if you want to be an engineer, but I must admit it sounds like a tremendously boring one to me.

Also, even if you make very certain that you're getting close to the actual result (taking a very small step size) this can introduce problems. The first is that even for a computer extremely small step sizes can lead to long computation times, but the other problem can be more pernicious. The computer rounds, and this rounding inevitably introduces some error. So, if you're dealing with very small step sizes, you're dealing with very small numbers, and these numbers are frequently rounded, which introduces errors that can accumulate over time. In fact, problems of this nature led to the first results in what we now call "chaos theory".

## 2 Improved Euler's Method

Euler's method is a pretty simple algorithm, and so as you might imagine it's not terribly hard to improve upon it. Here's we'll introduce an improvement upon Euler's method, and go over some results (but not derivations) of the error bounds we get when using these methods.

### 2.1 Error Bounds on Euler's Method

First, a theorem that gives us an error bound upon Euler's method, and lets us now how much we can improve our estimates if we decrease the value of our step size  $h$ .

#### Theorem

Suppose that the initial value problem

$$\begin{aligned}\frac{dy}{dx} &= f(x, y) \\ y(x_0) &= y_0\end{aligned}$$

has a unique solution  $y(x)$  on the closed interval  $[a, b]$  with  $a = x_0$ , and assume that  $y(x)$  has continuous second derivatives on  $[a, b]$ . Then there

exists a constant  $C$  such that the following is true: If the approximations  $y_1, y_2, \dots, y_k$  to the actual values  $y(x_1), y(x_2), \dots, y(x_k)$  at points of  $[a, b]$  are computed using Euler's method with step size  $h > 0$ , then

$$|y_n - y(x_n)| \leq Ch$$

for each  $n = 1, 2, \dots, k$ .

Now, this constant  $C$  is tricky to determine (it depends on the maximum value of  $|y''(x)|$  on the interval  $[a, b]$ ), and so the take-away from this theorem is that if you decrease your step size by  $1/2$ , you've decreased your maximum possible error by  $1/2$ . We note that this theorem assumes you don't have any problems with rounding error.

## 2.2 An Improvement to Euler's Method

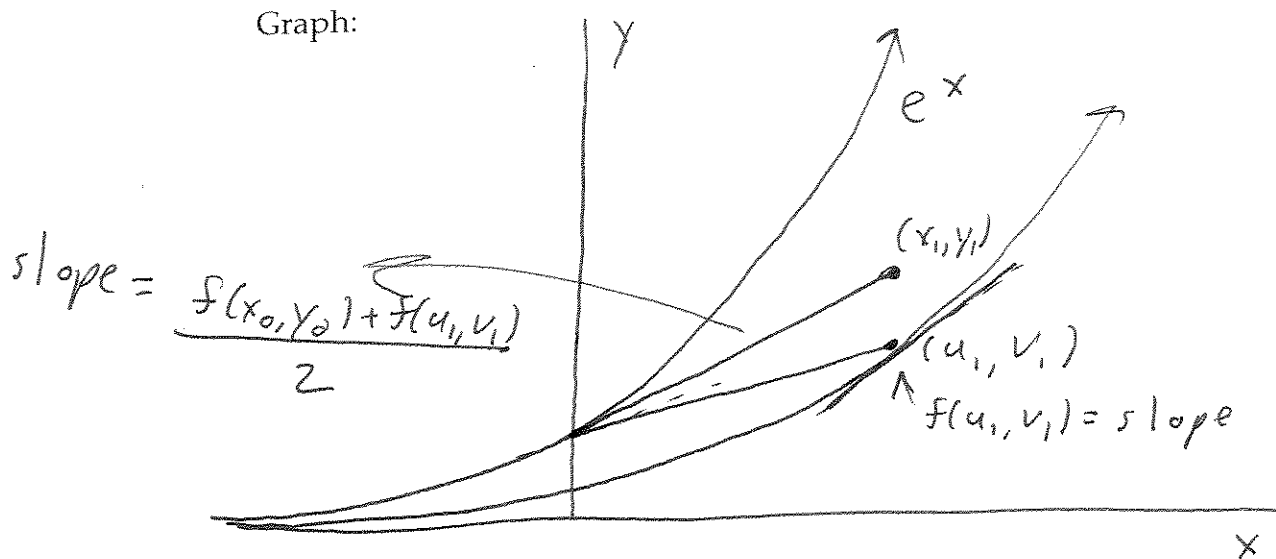
Our improvement on Euler's method is based upon getting a better estimate for what the slope is throughout the interval  $h$  upon which we're approximating it as being constant.

Here's the idea. We take our starting point  $(x_0, y_0)$  and we figure out what the slope is here  $f(x_0, y_0)$ , and we use this information, just as in Euler's method, to get a value for our next point:

$$\begin{aligned}u_1 &= x_0 + h \\v_1 &= y_0 + h \times f(x_0, y_0)\end{aligned}$$

We then calculate what the slope will be at the point  $(u_1, v_1)$ . We then use the *average* of the slopes  $f(x_0, y_0)$  and  $f(u_1, v_1)$  to calculate our next point  $(x_1, y_1)$ :

$$\begin{aligned}x_1 &= x_0 + h \\y_1 &= y_0 + h \left( \frac{f(x_0, y_0) + f(u_1, v_1)}{2} \right)\end{aligned}$$



We then repeat this algorithm to figure out the points  $(x_n, y_n)$ .

### 2.3 Error Bounds on Improved Euler's Method

The improved Euler's method involves more calculations, and so for it to be any use at all it had better give us better estimates. Well, it turns out that we have a nice theorem that says it does:

**Theorem**

Suppose that the initial value problem

$$\frac{dy}{dx} = f(x, y)$$

$$y(x_0) = y_0$$

has a unique solution  $y(x)$  on the closed interval  $[a, b]$  with  $a = x_0$ , and assume that  $y(x)$  has continuous second derivatives on  $[a, b]$ . Then there exists a constant  $C$  such that the following is true: If the approximations  $y_1, y_2, \dots, y_k$  to the actual values  $y(x_1), y(x_2), \dots, y(x_k)$  at points of  $[a, b]$  are computed using the improved Euler's method with step size  $h > 0$ , then

$$|y_n - y(x_n)| \leq Ch^2$$

for each  $n = 1, 2, \dots, k$ .

So, for example, if we halved our step size using Euler's method we'd only decrease our maximum error by one half, but if we halve our step size using the improved Euler's method, we decrease our maximum error by one fourth. Pretty sweet!