Error Analysis for Euler's Method

Math 131: Numerical Analysis

J.C. Meza

April 23, 2024

Introduction

Introduction

In order to discuss the error in Euler's method as well as its convergence, we will need to define a few terms.

First let's recall that we are seeking to approximate the solution to the IVP at a set of discrete points in time or ***mesh points*** typically of the form:

$$t_i = a + ih,\ i = 0, 1, 2, \dots, N.$$

We start with a few definitions.

Definitions

The ***difference method***

$$\begin{aligned} y_0 &= \alpha \\ y_{i+1} &= y_i + h\phi(t_i, y_i), \quad i = 0, 1, \ldots, N-1 \end{aligned}$$

has ***local truncation error***

$$d_{i+1} = \frac{y_{i+1} - (y_i + h\phi(t_i, y_i))}{h}, \tag{1}$$

where y_i denotes the solution of the difference equation at t_i, and $\phi(t, y)$ is a given function.

We call d_{i+1} ***local*** because it measures the accuracy of the solution at a specific point (step) in time. It is the error in 1 step if all previous values were exact and there's no roundoff error.

Definitions (cont.)

We say that a method is ***consistent (or accurate) of order*** q if q is the lowest positive integer such that

$$\max_i \; |d_i| = O(h^q). \tag{2}$$

Finally, the ***global error*** is defined as

$$e_i = y(t_i) - y_i \quad i = 0, 1, \ldots N, \tag{3}$$

where $y(t_i)$ is the true solution at time, t_i.

Gloabl error can be $>$ sum of local errors if ODE is unstable, or it could be $<$ sum of local errors if ODE is stable.

Example

Example

Show that Euler's method has local truncation error of $O(h)$.

For Euler's method $\phi(t_i, y_i) = f(t_i, y_i)$. As such we can write the local truncation error as:

$$
\begin{aligned}
d_{i+1} &= \frac{y_{i+1} - (y_i + f\phi(t_i, y_i))}{h} = \frac{\frac{h^2}{2} y''(\xi_i)}{h}, \\
&= \frac{h}{2} y''(\xi_i), \quad \xi_i \in (t_i, t_{i+1}),
\end{aligned} \tag{4}
$$

where the second equation is as a result of Taylor's theorem.

(cont.)

If we assume that the second derivative of y is bounded by some constant M, then we have:

$$|d_{i+1}| \leq \frac{M}{2} h \implies d_{i+1} = O(h)$$

By Equation 2 Euler's method is ***first order accurate***, i.e $q = 2$.

Remarks

- This example shows that the local truncation error for Euler's method is $O(h)$.
- Notice also that the error will depend on 1) the ODE, and 2) the step size.
- By the same argument, it is easy to see that Backward Euler is also first order accurate.

We now state a theorem that provides error bounds on the approximations generated by Euler's method and requirements for convergence.

Section 2

Convergence Theorem

Convergence and Global Error Estimates

Theorem (Euler Method Convergence.)

Suppose $f(t, y)$ is continuous and Lipschitz continuous in y, with constant L on a region $D = \{(t, y) |\ a \leq t \leq b,\ -\infty < y < \infty\}$. Let $y_1, \ldots, y_N$ be approximations generated by Euler's method for some integer $N > 0$.
***Then Euler's method converges and its global error decreases linearly in** h.*
Furthermore if a constant M exists with $|y''| \leq M,\ \forall t \in [a, b]$, then the global error satisfies

$$|e_i| \leq \frac{hM}{2L} \left[e^{L(t_i - a)} - 1\right] \ \forall i = 0, 1, 2, \ldots, N \tag{5}$$

Interpretation

Let's take a quick look to see what the error bound is saying.

Note that:

- The bound is exactly zero for $t_i = a$, which makes sense since $y_0 = y(a)$, the given initial condition.
- The last term depends on the Lipschitz constant, as well as the term $t_i - a$. But $t_i - a$ is bounded by $b - a$, so the entire term is also bounded.
- The bound depends on both the Lipschitz constant as well as the bound, M, on the second derivative of $y(t)$.
- ***the error bound is linear in*** h.

Key Points

- The good news is that we can bound the error at each time step.
- Nonetheless it is clear that the error bound will increase at each time step t_i.
- Our hope is that by choosing a small enough $h(\Delta t)$ we can compensate for the other terms and make the error bound small enough to generate an accurate approximation to $y(t)$.

Remark

Note that the theorem requires a bound on the second derivative. We can sometimes use knowledge of the partial derivatives to obtain an error bound. The important aspect is that the ***error bounds are linear in*** h*. Not surprisingly, as the number of computations grow so will the roundoff error.*

Section 3

Roundoff Error Analysis (Highlights)

Roundoff Error Analysis

As in the numerical differentiation lectures, we can derive a bound on the roundoff error:

$$|y(t_i) - y_i| \leq \frac{1}{L}\left(\frac{hM}{2} + \frac{\delta}{h}\right)\left[e^{L(t_i - a)} - 1\right] + |\delta_0| e^{L(t_i - a)},$$

where δ, δ_0 are constants representing the amount of roundoff error incurred at each time step.

Notice that we have the same situation as before with numerical differentiation – one of the terms is going to 0 while the second term blows up as $h \to 0$.

$$\text{error} \approx \left(\frac{hM}{2} + \frac{\delta}{h}\right).$$

Roundoff Error Analysis (Optimal h)

- A similar type of calculation as in the case of numerical differentiation, yields an optimal h:

$$h = \sqrt{\frac{2\delta}{M}}$$

that will depend on both δ and M.

- If we assume that $\delta \approx \epsilon$, i.e machine epsilon, then depending on the value of M, this implies h should be roughly the square root of machine epsilon.

- For IVPs, the more important question is ***stability***, which will depend on choosing an appropriate h.

Stability

- While we can usually get a more accurate solution as h decreases, that means we will also necessarily increase the computational cost.
- If we increase h, however we run against another problem - one of stability
- It can be shown that for forward Euler, there is an upper bound for h determined by a condition known as ***absolute stability***.
- This requirement does not hold for Backward Euler!
- Unfortunately, we don't have time to cover that topic here.

Brief Introduction to Stability

Consider the test equation

$$y' = \lambda y, y(0) = y_0$$

for which the solution is given by $y(t) = e^{\lambda t} y(0)$.

- for $\lambda > 0$ the exact solution increases, for $\lambda < 0$ the exact solution decreases.
- For Euler's method it would make sense that $|y_{i+1}| \leq |y_i|$
- Given that $y_{i+1} = (1 + h\lambda) y_i$ it can be shown that this requires that:

$$h \leq \frac{2}{|\lambda|}$$

Estimating the order of a method

- Suppose the error term is given by $e(h) \approx Ch^q$, C independent of h.
- As before, we can then estimate the error at $2h$, so that $e(2h) \approx C(2h)^q \approx 2^q e(h)$.
- We can the approximate the rate, q, by:

$$\text{rate} = \log_2 \left(\frac{e(2h)}{e(h)} \right)$$

Demo

Section 5

Supplementary Materials (Proofs)

Convergence for Euler's Method

We had to bypass the proof of the convergence of Euler's method. It is not a difficult proof and it uses standard techniques. If you're interested this section will provide a brief overview of the proof.

First, we will need a few lemmas that are used in the proof of the convergence of Euler's method. They are included here for completeness.

Lemma 5.7

Lemma 5.7. For all $x \geq 1$ and any positive m we have

$$0 \leq (1+x)^m \leq e^{mx} \tag{6}$$

Proof. Straightforward application of Taylor's Theorem to $f(x) = e^x$ about $x_0 = 0$.

Lemma 5.8

Lemma 5.8. If

- s, t are positive real numbers
- $\{a_i\}_{i=0}^{k}$ is a sequence satisfying $a_0 \geq -\frac{t}{s}$
- $a_{i+1} \leq (i+s)a_i + t \quad i = 0, 1, \ldots, k-1$

Then $a_{i+1} \leq e^{(i+1)s}(a_0 + \frac{t}{s}) - \frac{t}{s}$.

Proof. Left as an exercise. If you're interested in trying to prove it, the idea is to use a geometric series to show that

$$a_{i+1} \leq (i+s)^{i+1}(a_0 + \frac{t}{s}) - \frac{t}{s}$$

followed by an application of Equation 6, with $x = s$ to show result.

Convergence for Euler's method Theorem 2.1.

- A method is said to *converge* if the maximum global error tends to 0 as h tends to 0 (assuming that an exact solution exists and is sufficiently smooth.
- For Euler' method, which is $O(h)$, we would then expect that $e_i = y(t_i) - y_i$ should be of the same order.
- Let's consider the local truncation error first:

$$d_i = \frac{y(t_{i+1}) - y(t_i)}{h} - f(t_i, y(t_i)),$$
$$0 = \frac{y_{i+1} - y_i}{h} - f(t_i, y_i),$$

(cont.)

Subtracting the two, gives us a difference formula for the local truncation error:

$$d_i = \frac{e_{i+1} - e_i}{h} - [f(t_i, y(t_i)) - f(t_i, y_i)]$$

Solving for the error at t_{i+1}, we have:

$$e_{i+1} = e_i + h\,[f(t_i, y(t_i)) - f(t_i, y_i)] + hd_i, \quad i = 0, \ldots, N$$

(cont.)

Taking absolute values:

$$|e_{i+1}| \leq |e_i| + h|\,[f(t_i, y(t_i)) - f(t_i, y_i)]\,| + |hd|,$$

where d is the maximum of $|d_i|$ over all time steps.

Since f is Lipschitz continuous with constant L, we can simplify the error difference equation to:

$$\begin{aligned} |e_{i+1}| &\leq |e_i| + hL|e_i| + |hd|, \\ &\leq (1 + hL)|e_i| + hd. \end{aligned}$$

(cont.)

Now note that we can repeat this process with e_i, which would gives us:

$$
\begin{aligned}
|e_{i+1}| &\leq (1 + hL)|e_i| + hd, \\
&\leq (1 + hL)[(1 + hL)|e_{i-1} + hd| + hd = (1 + hL)^2|e_{i-1}| + (1 + hL) \\
&\leq \cdots \leq (1 + hL)^{i-1}|e_0| + hd \sum_{j=0}^{i} (1 + hL)^j \\
&\leq d \left[e^{L(t_i - a)} - 1 \right] / L
\end{aligned}
$$

where we've used the Lemmas above to compute the sum and the fact that $e_0 = 0$.

(cont.)

The final step is to note that by definition of we have that:

$$d \geq \max_{0 \leq i \leq N-1} |d_i|$$
$$d_i = \frac{h}{2} y''(\xi_i) \quad i = 0, \ldots, N$$

So if we can bound the second derivative such that:

$$M = \max_{a \leq t \leq b} |y''(t)| \implies d = \frac{h}{2} M.$$

which gives us our desired error bound:

$$|e_i| \leq \frac{Mh}{2L} \left[e^{L(t_i - a)} - 1 \right], \quad i = 0, 1, \ldots, N.$$