NSDE 1: LECTURE 1

TYRONE REES*

Examples in presentation:

- Moon landing and code
- Finding Dory
- Driverless cars

In the examples we've seen, and most other 'real world' systems, the differential equations cannot be solved analytically, and instead they must be solved approximately using numerical algorithms.

Questions that this course will answer:

- **How** do I solve an initial value problem (IVP) for an ordinary differential equation (ODE), or an IVP for a parabolic partial differential equation?
- **How** do I analyse the accuracy of the solution I get?
- **How** can I know if the algorithm I'm using is stable?

1. Initial value problems for ODEs. Here is the basic form that we will use throughout the course:

$$\begin{aligned} \mathbf{u}' &= \mathbf{f}(t, \mathbf{u}), \qquad \text{for } t \in [t_0, T_M] \\ \mathbf{u}(t_0) &= \mathbf{u}_0 \end{aligned}$$

where $\mathbf{u}, \mathbf{u}_0, \mathbf{f}(\cdot) \in \mathbb{R}^k$ for some k.

First we need to know: is this a sensible question to ask?

EXAMPLE 1.1.

$$\begin{aligned} u' &= 3u^{2/3} \\ u(0) &= 0. \end{aligned}$$

What are the solutions to this problem?

A-level maths tells us that

$$\begin{aligned} du/dt &= 3u^{2/3} \\ \int \frac{1}{3} u^{-2/3}\, du &= \int dt \\ u^{1/3} &= t + K \\ u &= (t+K)^3 \end{aligned}$$

and, since $u(0) = 0$, we have that $u = t^3$.

However, *there is another solution – $u = 0$!*

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

In fact, it's worse than that – as

$$u(t) = \begin{cases} 0 & 0 \leq t \leq c \\ (t-c)^3 & c < t < \infty \end{cases}$$

is a solution for all c – **an infinite number of solutions!**.

This spells trouble for an algorithm designed to find an approximation to the solution – which one would it pick?!

One way to guarantee that we don't have this problem is to ensure that the problem is smooth enough. Let us, for simplicity, consider the 1D problem,

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

This is what we mean by 'smooth enough'. Let (t_0, u_0) be the point of the initial data. We need the problem to satisfy two conditions:

- We can draw 'butterfly wings' from (t_0, u_0) the function f stays within this region.
- We can draw a box around the solution in some region, and the solution leaves the box from the edge (not the top).

If these conditions hold, then we can be sure that a solution exists, and that it is unique.

We state these conditions more rigorously:

THEOREM 1.2. **Picard's theorem**

Let $R = \{z \in \mathbb{R}^2 : |t - t_0| \leq h, |u - u_0| \leq k\}$

- *Suppose that* $f(\cdot,\cdot)$ *is a continuous function in* $U \supset R$ *with*

$$M = \max_{(t,u)\in R} |f(t,u)|$$

- *Suppose that* $\exists L > 0$ *s.t.* $\forall (t,u_1),(t,u_2) \in R$,

$$|f(t,u_1) - f(t,u_2)| \leq L|u_1 - u_2|$$

Lipschitz condition

- *Suppose that* $Mh \leq k$

Then there exists a unique continuously differentiable function $t \to u(t)$ *satisfying the IVP:*

$$u' = f(t,u), \qquad u(t_0) = u_0$$

for all $t \in [t_0 - h, t_0 + h]$.

NSDE 1: LECTURE 2

TYRONE REES*

Recap

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

No guarantee of a unique solution. However, there is one if the conditions for **Picard's theorem** are satisfied:

Let $R = \{z \in \mathbb{R}^2 : |t - t_0| \le h, |u - u_0| \le k\}$

- Suppose that $f(\cdot,\cdot)$ is a continuous function in $U \supset R$ with

$$M = \max_{(t,u)\in R} |f(t,u)|$$

- Suppose that $\exists L > 0$ s.t. $\forall (t,u_1),(t,u_2) \in R$,

$$|f(t,u_1) - f(t,u_2)| \le L|u_1 - u_2|$$

Lipschitz condition

- Suppose that $Mh \le k$

Then there exists a unique continuously differentiable function $t \to u(t)$ satisfying the IVP:

$$u' = f(t,u), \qquad u(t_0) = u_0$$

for all $t \in [t_0 - h, t_0 + h]$.

EXAMPLE 0.1.

$$u' = 3u^{2/3}, \qquad u(0) = 0$$

We know from last time that there is not a unique solution, so we should expect that u violates a condition of Picard's theorem.

Suppose there exist constants $h > 0$ and $k > 0$ so that, if $|t| \le h$, $|u| \le k$, there is a constant $L > 0$ such that

$$|f(t,u_1) - f(t,u_2)| \le L|u_1 - u_2|.$$

Then, if we set $u_1 = u \in [0,k]$, $u_2 = 0$, then we have that

$$|3u^{2/3}| = 3|u|^{2/3} < L|u| \iff |u| > (L/3)^3$$

This is clearly a contradiction, as u can be arbitrarily close to zero.

```
ezplot('3*x^(3/2)',[0,2]);
```

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

Before we dive into an example that is unique, we remind ourselves of a useful result:

THEOREM 0.2. **The Mean Value Theorem** *Suppose that a function g is defined and continuous on a closed interval $[a, b]$ on the real line, and that g is differentiable on the open interval (a, b).*

Then, for every $y_1, y_2 \in [a, b]$, there exists $\xi \in (y_1, y_2)$ such that

$$g(y_1) - g(y_2) = g'(\xi)(y_1 - y_2).$$

This is very useful for deriving Lipschitz constants!

EXAMPLE 0.3.

$$u' = u^2 \qquad u(0) = u_0$$

First, let us define the region

$$R = \{(t, u) \in \mathbb{R}^2 : |t| \leq h, |u - u_0| \leq k\}$$

for some $h > 0$ and $k > 0$.

Note that

$$\begin{aligned} |f(t, u_1) - f(t, u_2)| &= |u_1^2 - u_2^2| \\ &\leq 2(u_0 + k)|u_1 - u_2| \qquad \textit{(by MVT)} \end{aligned}$$

So $f(\cdot, \cdot)$ has a Lipschitz constant of $2(u_0 + k)$ in any neighbourhood R, for any h and k.

Also, $\max |f(t, u)| = (k+u_0)^2$, so given any $k > 0$, taking $h \leq k/(k+u_0)^2$ will ensure that the conditions of Picard's theorem are satisfied.

We must be careful about what Picard's theorem says, though. The analytic solution here is

$$u = \frac{u_0}{1 - u_0 t},$$

which blows up at time $t = 1/u_0$.

```
ezplot('a/(1-a*x)',x=[0,1/a])
```

1. Solving ODEs numerically. How do we approximate the solution of ODEs?

Recall our standard problem:

$$u' = f(t, u), \text{ for } t \in [t_0, T_M], \qquad u(t_0) = u_0.$$

First, we seek approximate the solution at a discrete number of points in time:

Since we have the inital value, this will be exact, but the other points only approximate the true solution.

Notation

We approximate the solution of the ODE at points that we call

$$t_0, t_1, \ldots, t_m.$$

We will often (for simplicity) consider equally spaced points, so that

$$t_{n+1} = t_n + h, \text{ where } h = (t_n - t_0)/n.$$

The exact solution at these points would be

$$u(t_0), u(t_1), \ldots, u(t_m).$$

We call the approximate solution at these points

$$U_0, U_1, \ldots, U_m.$$

How can we approximate the solution?

$$\begin{aligned} u' &= f(t,u) \\ \lim_{t\to 0} \frac{u(t+h)-u(t)}{h} &= f(t,u) \\ \frac{u(t+h)-u(t)}{h} &\approx f(t,u) \\ u(t_n+h) &\approx u(t_n) + hf(t_n, u(t_n)) \end{aligned}$$

We can turn this approximation into an algorithm:

Euler's (explicit) method

$$\begin{aligned} U_0 &= u(t_0) \\ U_{n+1} &= U_n + h\, f(t_n, U_n) \end{aligned}$$

Systems

We developed this for first-order IVPs, but the method works more generally. For example, consider the system

$$\begin{aligned} u'' &= f(t, u, u') \\ u(0) &= u_0 \\ u'(0) &= u_0'. \end{aligned}$$

This can be re-written as a system of first order equations:

$$\begin{aligned} u' &= v \qquad v(0) = u_0' \\ v' &= f(t,u,v) \qquad u(0) = u(0), \end{aligned}$$

or, in vector form:

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}, \qquad \mathbf{u}(0) = \begin{bmatrix} u_0 \\ u_0' \end{bmatrix}$$

where

$$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix} \text{ and } \mathbf{f} = \begin{bmatrix} v \\ f(t,u,v) \end{bmatrix}$$

We can then apply Euler's method, say, in exactly the same way as for the scalar case:

$$\mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{f}(t_n, \mathbf{U}_n).$$

This is just shorthand for writing Euler separately on each of the components:

$$\begin{aligned} U_{n+1} &= U_n + h\, V_n \\ V_{n+1} &= V_n + h\, f(t_n, U_n, V_n) \end{aligned}$$

NSDE 1: LECTURE 3

TYRONE REES*

Recap

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

Consider a grid of $m+1$ equally spaced points, $t_{n+1} = t_n + h$. Approximate the solution $u(t_0)$, $u(t_1), \ldots, u(t_m)$ by U_0, $U_1, \ldots, U_m$, which are calculated by

$$\begin{aligned} U_0 &= u(t_0) \\ U_{n+1} &= U_n + hf(t_n, U_n)) \end{aligned}$$

We can think of Euler's method in terms of integrals:

$$\begin{aligned} \int_{t_n}^{t_{n+1}} u'\,dt &= \int_{t_n}^{t_{n+1}} f(x,u)\,dt \\ u(t_{n+1}) - u(t_n) &= \int_{t_n}^{t_{n+1}} f(t,u)\,dt \\ u(t_{n+1}) &= u(t_n) + \int_{t_n}^{t_{n+1}} f(t,u)\,dt \\ u(t_{n+1}) &\approx u(t_n) + h\, f(t_n, u(t_n)) \end{aligned}$$

However, we can also make a different approximation:

$$\begin{aligned} u(t_{n+1}) &= u(t_n) + \int_{t_n}^{t_{n+1}} f(t,u)\,dt \\ u(t_{n+1}) &\approx u(t_n) + h\, f(t_{n+1}, u(t_{n+1})) \end{aligned}$$

Which leads to the algorithm:

Euler's implicit method

$$\begin{aligned} U_0 &= u(t_0) \\ U_{n+1} &= U_n + h\, f(t_{n+1}, U_{n+1}) \end{aligned}$$

This has an obvious problem: U_{n+1} appears on the right hand side of the equation. Sometimes (e.g., on problem sheets) such problems can be solved straightforwardly, but usually, in practice, we'll need to use a nonlinear equation solver (e.g., Newton's method).

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

This is why this method is called **implicit**.

We could also do a more sensible (from the point of view of approximating the integral) approximation of

$$\int_{t_n}^{t_{n+1}} f(t,u)\,dt \approx \frac{h}{2}(f(t_n,u(t_n)) + f(t_{n+1},u(t_{n+1})))$$

This motivates what's known as the **Trapezium rule method**:

$$\begin{aligned} U_0 &= u(t_0) \\ U_{n+1} &= U_n + \frac{h}{2}(f(t_n,U_n) + f(t_{n+1},U_{n+1})) \end{aligned}$$

And these are all specific instances of a θ**-method**:

$$\begin{aligned} U_0 &= u(t_0) \\ U_{n+1} &= U_n + h\left((1-\theta)f(t_n,U_n) + \theta f(t_{n+1},U_{n+1})\right) \end{aligned}$$

which takes a **weighted** average of the end points to approximate the integral.

Example 0.1.

$$y' = x - y^2, \qquad y(0) = 0$$

Solve using a θ-method for $\theta = 0, 1/2,$ and 1.

Accuracy

How can we know beforehand how accurate the method is going to be? Before we can answer this, we need a concept of error.

The **global error** of any method is defined as

$$e_n = u(t_n) - U_n.$$

We can also define the **truncation error**. Note that the methods we've looked at so far can be written in the form

$$U_{n+1} = U_n + h\Phi(t_n, t_{n+1}, U_n, U_{n+1}; h)$$

We can re-write this in a form that explicitly models the derivatives:

$$\frac{U_{n+1} - U_n}{h} = \Phi(t_n, t_{n+1}, U_n, U_{n+1}; h).$$

The trunction error is the difference between the left and right hand sides *if we plug in the exact solution:*

$$T_n = \frac{u(t_{n+1}) - u(t_n)}{h} - \Phi(t_n, t_{n+1}, u(t_n), u(t_{n+1}); h)$$

If $T_n = O(h^p)$, where p is the largest such integer: the method is said to be p**th order accurate**.

EXAMPLE 0.2. **Euler's method: truncation error**

$$\begin{aligned} T_n &= \frac{u(t_{n+1}) - u(t_n)}{h} - f(t_n, u(t_n)) \\ &= \frac{u(t_{n+1}) - u(t_n)}{h} - u'(t_n) \end{aligned}$$

We can expand $u(t_{n+1})$ via a Taylor series to give

$$u(t_{n+1}) = u(t_n + h) = u(t_n) + hu'(t_n) + \frac{h^2}{2}u''(\xi_n), \qquad \xi_n \in [t_n, t_{n+1}]$$

Subsituting this in we get

$$T_n = \frac{u(t_n) + hu'(t_n) + \frac{h^2}{2}u''(\xi_n) - u(t_n)}{h} - u'(t_n) = \frac{1}{2}hu''(\xi_n)$$

Euler's method is first order accurate.

The truncation error is usually 'easy' to calculate, but it tells us little, by itself, about the quality of the solution. What we really want to know is the size of the global error. Now,

$$\begin{aligned} u(t_{n+1}) &= u(t_n) + hf(t_n, u(t_n)) + hT_n \\ U_{n+1} &= U_n + hf(t_n, U_n) \\ \hline e_{n+1} &= e_n + h[f(t_n, u(t_n)) - f(t_n, U_n)] + hT_n \end{aligned}$$

and, taking absolute values of both sides:

$$|e_{n+1}| \le |e_n| + h|f(t_n, u(t_n)) - f(t_n, U_n)| + h|T_n|$$

If $f(\cdot, \cdot)$ satisfies a Lipschitz condition, then $|f(t_n, u(t_n)) - f(t_n, U_n)| \le L|u(t_n) - U_n|$, and so we have

$$|e_{n+1}| \le (1 + Lh)|e_n| + h|T_n| \qquad \text{for all } n$$

Now if we let $T = \max |T_n|$, we have

$$|e_{n+1}| \le (1 + Lh)|e_n| + hT \qquad \text{for all } n$$

By induction, we get that

$$|e_n| \le (1 + Lh)^n|e_0| + \frac{T}{L}[(1 + Lh)^n - 1] \qquad \text{for all } n$$

We know that $e_0 = 0$, and also that

$$(1 + Lh)^n \le [e^{Lh}]^n = e^{Lhn} = e^{L(t_n - t_0)},$$

where we have used the facts that

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots \geq 1 + x \qquad \text{for } x > 0$$

and that

$$t_n = t_0 + n\,h.$$

Therefore we get

$$|e_n| \leq \frac{T}{L}\left[e^{L(t_n - t_0)} - 1\right]$$

Finally, note that

$$T = \max_n |T_n| = \max_n \frac{h}{2}|u''(\xi_n)| \leq \frac{1}{2}hM_2$$

where we have set $M_2 = \frac{1}{2}\max_{t\in[t_n,t_{n+1}]}|u''(t)|$. We therefore have that

$$|e_n| \leq \frac{M_2}{2L}(e^{L(t_n - t_0)} - 1)h \leq \text{Const.}h$$

Hence, if you half the step size, you (at least) half the error.

NSDE 1: LECTURE 4

TYRONE REES*

Recap

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

We defined the **global error** as

$$e_n = u(t_n) - U_n$$

and the truncation error as

$$T_n = \frac{(t_{n+1} - u(t_n))}{h} - \Phi(t_n, t_{n+1}, U_n, U_{n+1}; h).$$

For Euler's method:

$$T_n - \frac{h}{2}u''(\xi_n)$$

We saw last time that

$$|e_n| \leq \frac{T}{L}\left[e^{L(t_n - t_0)} - 1\right]$$

Finally, note that

$$T = \max_n |T_n| = \max_n \frac{h}{2}|u''(\xi_n)| \leq \frac{1}{2}hM_2$$

and so

$$|e_n| \leq \frac{M_2}{2L}(e^{L(t_n-t_0)} - 1)h \leq \text{Const.}h$$

`euler_half_step.m`

Hence, if you half the step size, you (at least) half the error.

θ-methods

A similar (but more messy) analysis can be done for $\theta-$methods. Here we get that

$$|e_n| \leq \frac{h}{L}\left\{\left|\frac{1}{2} - \theta\right| M_2 + \frac{1}{3}hM_3\right\}\left[\mathrm{e}^{\frac{L(t_n-t_0)}{1-\theta Lh}} - 1\right],$$

where $M_3 = \max_{t\in[t_0,t_m]} |u'''(t)|$.

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

As expected, the error of explicit Euler is recovered if we take $\theta = 0$. We also see that the special choice of $\theta = 1/2$ – the trapezium rule method – gives second order accurate method. This explains the behaviour in the numerical example.

Question: where did M_3 go for Euler's method?

Recall that when we derived the truncation error, we cut off the Taylor series at the u'' term. We could equally have written

$$T_n = \frac{h}{2}u''(t_n) + \frac{h^2}{6}u'''(\hat{\xi}_n), \qquad \hat{\xi} \in [t_n, t_{n+1}.$$

Check that, if we'd included this term, we obtain the bound for the θ-method.

One-step methods

Euler's method is an example of what's known as a one-step method. The general form of a one-step method is:

$$\begin{aligned} U_0 &= u(t_0) \\ U_{n+1} &= U_n + h\Phi(t_n, U_n; h) \end{aligned}$$

Note that all that appears in the right hand side is U_n.

Examples

Euler's method:

$$\Phi(t_n, U_n; h) = f(t_n, U_n)$$

The trapezium rule is **not** a one-step method:

$$\Phi = \frac{1}{2}\left(f(t_n, U_n) + f(t_{n+1}, U_{n+1})\right)$$

We can, however, get what should be a better method, which is **explicit**, by replacing U_{n+1} itself by it's Euler approximation, $U_n + hf(t_n, U_n)$. This is called **improved Euler's method**.

`improved_euler.m`

For any one-step method, if Φ satisfies a Lipschitz condition:

$$|\Phi(t, u; h) - \Phi(t, v; h)| \leq L|u - v|,$$

then the same analysis we did for Euler's method will go through here also. We therefore also have the concept of **Truncation error** here:

$$T_n = \frac{u(t_{n+1}) - u(t_n)}{h} - \Phi(t_n, u(t_n); h),$$

and, as before, if $T = \max_n |T_n|$, then

$$|U_n - u(t_n)| \leq (e^{L(t_n - t_0)} - 1)\frac{T}{L}$$

In order for the error to vanish with a small enough step size, we therefore require that $T_n \to 0$ as $h \to 0$ and $n \to \infty$, with $nh = t_n - t_0$.

Consistency: Definition We say that a one-step method is **consistent** if

$$\lim_{h\to 0, n\to\infty, nh=t_n-t_0} T_n = 0$$

Now, since $\Phi(\cdot,\cdot;\cdot)$ and $y'(\cdot)$ are continuous, we know that

$$\lim_{h\to 0} T_n = u'(t_n) - \Phi(t_n, u(t_n); 0).$$

Therefore a one-step method is consistent if and only if

$$\Phi(t_n, u(t_n); 0)) = f(t, u).$$

In general, we are not restricted to one extra point in the region $[t_n, t_n + h]$; we can evaluate the function at as many intermediate points as we like. We're only interested in explicit methods, so we need to also have a way of approximating the solution at those points (as $u(t_n + ah)$ will not be avaliable). This leads to a general family of methods known as Runge-Kutta methods.

Runge-Kutta R-stage method

$$\begin{aligned}
U_{n+1} &= U_n + h\Phi(t_n, U_n; h) \\
\Phi(x, y; h) &= \sum_{r=1}^{R} c_r k_r \\
k_1 &= f(t_n, U_n) \\
k_r &= f\left(t_n + a_r\, h, U_n + h\sum_{s=1}^{r-1} b_{rs} k_s, \quad r = 2, \ldots, R\right) \\
a_r &= \sum_{s=1}^{r-1} b_{rs}, \quad r = 2, \ldots, R
\end{aligned}$$

This is usually written in the form of a Butcher table:

$$\begin{array}{c|c} a = B\mathbf{1} & B \\ \hline & c^T \end{array}$$

R=1

If $R = 1$, then we get back our old friend Euler's method.

R = 2

For $R = 2$, we have more choice. Now we want to find a method such that

$$U_{n+1} = U_n + h(c_1 k_i + c_2 k_2),$$

where

$$k_1 = f(t_n, U_n)$$
$$k_2 = f(t_n + a_2 h, U_n + b_{21} h k_1)$$

What values of c_1, c_2, a_2, and b_{21} make sense?

To be consistent, we need that $\Phi(t_n, U_n; h) = f(t_n, U_n)$, i.e.

$$c_1 f(t_n, U_n) + c_2 f(t_n, U_n) = f(t_n, U_n) \iff c_1 + c_2 = 1$$

So, given c_2, we can obtain c_1, but how should we choose c_2, b_{21} and a_2. We try to make the order of the method as high as possible.

Recall

$$\begin{aligned} T_n &= \frac{u(t_{n+1}) - u(t_n)}{h} - \Phi(t_n, u(t_n)) \\ &= \frac{u(t_{n+1}) - u(t_n)}{h} - c_1 f(t_n, u(t_n)) - c_2 f(t_n + a_2 h, u(t_n) + b_{21} f(t_n, u(t_n))h) \end{aligned}$$

By expanding $u(t_n+h)$ about t_n, and $f(t_n+a_2h, u(t_n)+b_{21}f(t_n, u(t_n)h))$ about t_n then $u(t_n)$, and noting that

$$\begin{aligned} u'(t_n) &= f(t_n, u(t_n)) \\ u''(t_n) &= \frac{d}{dt} f(t_n, u(t_n)) \\ &= \frac{\partial}{\partial t} f(t_n, u(t_n)) + \frac{\partial}{\partial u} f(t_n, u(t_n)) \frac{d}{dt} u(t_n) \\ &= \frac{\partial}{\partial t} f(t_n, u(t_n)) + f(t_n, u(t_n)) \frac{\partial}{\partial u} f(t_n, u(t_n)), \end{aligned}$$

we can show that

$$\begin{aligned} T_n = h^2 \Bigg(\frac{1}{6} u'''(t_n) - \frac{c_2}{2} \Bigg[& b_{21}^2 f(t_n, u(t_n)) \frac{\partial^2}{\partial u^2} f(t_n, u(t_n)) + \\ & a_2 b_{21} \frac{\partial^2}{\partial u \partial t} f(t_n, u(t_n)) + a_2^2 \frac{\partial^2}{\partial t^2} f(t_n, u(t_n)) \Bigg] \Bigg) + O(h^3) \end{aligned}$$

as long as we choose

$$1/2 = c_2 a_2 = c_2 b_{21}.$$

NSDE 1: LECTURE 5

TYRONE REES*

Recap

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

Runge-Kutta R-stage method

$$\begin{aligned} U_{n+1} &= U_n + h\Phi(t_n, U_n; h) \\ \Phi(t,u;h) &= \sum_{r=1}^{R} c_r k_r \\ k_1 &= f(t_n, U_n) \\ k_r &= f\left(t_n + a_r\, h, U_n + h\sum_{s=1}^{r-1} b_{rs} k_s, \quad r = 2,\dots,R\right) \\ a_r &= \sum_{s=1}^{r-1} b_{rs}, \quad r = 2,\dots,R \end{aligned}$$

This is usually written in the form of a Butcher table:

$$\begin{array}{c|c} a & B \\ \hline & c^T \end{array}$$

Last time we saw that, for $R = 2$, we require that the coefficient satisfy

$$1/2 = c_2 a_2 = c_2 b_{21},$$

which tells us we should take $b_{21} = a_2$, $c_2 = 1/2a_2$, and $c_1 = 1 - 1/(2a_2)$. We still have a free parameter, a_2, which can take any value and still give a second order method. (Note that no choice of parameters will, in general, give a third order method).

Popular choices are:

$a_2 = 1/2$**:**

$$\begin{array}{c|cc} 0 & 1 & \\ 1/2 & 1/2 & \\ \hline & 0 & 1 \end{array}$$

This gives:

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

$$U_{n+1} = U_n + hf(t_n + 1/2h, U_n + 1/2hf(t_n, U_n))$$

Which is a method called **Modified Euler**.

$a_2 = 1$**:**

$$\begin{array}{c|cc} 0 & 1 & \\ 1 & 1 & \\ \hline & 1/2 & 1/2 \end{array}$$

This gives:

$$U_{n+1} = U_n + \frac{h}{2}\left(f(t_n, U_n) + f(t_n + h, U_n + hf(t_n, U_n))\right)$$

Which is, of course, the method we started with, **improved Euler**.

R=3 The same trick can be done (with messier algebra) to obtain three stage Runge-Kutta method. Again, the consistency condition is that

$$c_1 + c_2 + c_3 = 1.$$

Now, we can obtain $T_n = O(h^3)$ if we choose the parameters to satisfy

$$\begin{aligned} c_2 b_{21} + c_2(b_{31} + b_{32}) &= \frac{1}{2} \\ c_2 b_{21}^2 + c_3(b_{31} + b_{32})^2 &= \frac{1}{3} \\ c_3 b_{21} b_{32} &= \frac{1}{6} \end{aligned}$$

Including the consistency condtion, we therefore have four equations for six unknowns, leaving two parameters free.

A few examples are important enough to have a name...

The *classical* RK method is:

$$\begin{array}{c|ccc} 0 & 1 & & \\ 1/2 & 1/2 & & \\ 1 & -1 & 2 & \\ \hline & & & \\ & 1/6 & 2/3 & 1/6 \end{array}$$

(which is related to Simpson's rule). The *Nystrom* scheme is:

$$\begin{array}{c|ccc} 0 & 1 & & \\ 2/3 & 2/3 & & \\ 2/3 & 0 & 2/3 & \\ \hline & 1/4 & 3/8 & 3/8 \end{array}$$

R=4

A widely used fourth order method has Butcher table:

$$\begin{array}{c|cccc} 0 & 1 & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array},$$

which corresponds to the method

$$\begin{aligned} U_{n+1} &= U_n + \frac{1}{6}h\left\{k_1 + 2k_2 + 2k_3 + k_4\right\} \\ k_1 &= f(t_n, U_n) \\ k_2 &= f(t_n + \frac{1}{2}h, U_n + \frac{1}{2}hk_1) \\ k_3 &= f(t_n + \frac{1}{2}h, U_n + \frac{1}{2}hk_2) \\ k_4 &= f(t_n + h, U_n + hk_3). \end{aligned}$$

`compare_methods.m`

Adaptive time steps. If the solution changes very slowly, then we may be able to get a pretty good approximation with a large time step. However, if the solution changes rapidly, we won't be able to resolve the details unles we use a sufficiently small time step.

We can use our knowledge of the error to inform us of where we should next approxmimate the solution. Since the step size will change, we'll use the notation $t_{n+1} = t_n + \Delta t_n$. If the error is too large, we can reduce it by taking a smaller step.

We'll see this by way of an example. For a fourth order Runge-Kutta method, our approxiation satisfies

$$u(t_{n+1}) = U^a_{n+1} + K_1(\Delta t_n)^5 u^{(v)}(t_n) + O(\Delta t_n^6)$$

for some constant K_1.

As well as a step size of Δt_n, we could also have taken two steps of size $\Delta t_n/2$ to give us a *different* approximation at the same point. The error here will satisfy:

$$u(t_{n+1}) = U^b_{n+1} + 2K_1(\Delta t_n/2)^5 u^{(v)}(t_n) + O(\Delta t_n^6)$$

(convince yourself this is true – i.e., that the constants are the same in both cases).

Subtracting these gives

$$|U^b_{n+1} - U^a_{n+1}| \approx \frac{15}{16} K_1$$

Now, suppose we wanted this difference to take some value, ϵ. How would we pick a time step $\bar{\Delta t}_n$ to ensure this?

Suppose that

$$\epsilon = K_1 (\bar{\Delta t}_n)^5 u^{(v)}(t_n),$$

and so we can remove the unknowns by dividing these two expressions, giving:

$$\left(\frac{\bar{\Delta t}_n}{\Delta t_n} \right)^5 = \frac{\epsilon}{|U^b_{n+1} - U^a_{n+1}|}$$

Rearranging, we get that we should take

$$\bar{\Delta t}_n = \left(\frac{\epsilon}{|U^b_{n+1} - U^a_{n+1}|} \right)^{1/5} \Delta t_n.$$

This gives us a method for adapting the step length.

- if $\bar{\Delta t}_n < \Delta t_n$, (i.e. $|U^b_{n+1} - U^a_{n+1}| > \epsilon$) repeat the step from t_n with the reduced step length
- if $\bar{\Delta t}_n > \Delta t_n$, (i.e. $|U^b_{n+1} - U^a_{n+1}| \leq \epsilon$), take $U_{n+1} = U^b_{n+1}$ and set $\Delta t_{n+1} = \bar{\Delta t}_n$.

This gives a method of adapting the step length depending on the properties of the equation being solved. However, it is more expensive – at each step we do an extra application of RK4.

NSDE 1: LECTURE 6

TYRONE REES*

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

Last time we looked at ways of controlling the step size using adaptive algorithms. This is useful for when we want the error to remain within some tolerance.

Another strategy is to use Runge-Kutta methods of different orders on top of each other. Because of the flexibility allowed in RK methods, it is possible to find sets of two families that evaluate the function at the same points, yet have different orders. For example, consider the method with Butcher table:

$$\begin{array}{c|cccc} 0 & 1 & & & \\ 1/2 & 1/2 & & & \\ 3/4 & 0 & 3/4 & & \\ 1 & 2/9 & 3/9 & 4/9 & \\ \hline & 2/9 & 3/9 & 4/9 & \\ & 11/72 & 30/72 & 40/72 & -9/72 \end{array},$$

which is short-hand for the two methods

$$U^a_{n+1} = U_n + \frac{\Delta t_n}{9}(2k_1 + 3k_2 + 4k_3)$$

(a second order method), and

$$U^b_{n+1} = U_n + \frac{\Delta t_n}{72}(11k_1 + 30k_2 + 40k_3 - 9k_4)$$

(a third order method).

Here we can say that $u(t_{n+1}) = U^a_{n+1} + K_2(\Delta t_n)^2 + O(\Delta t_n)^3$, and also that $u(t_{n+1}) = U^b_{n+1} + O(\Delta t_n)^3$. Since, for small Δt_n, we can assume that the $O(\Delta t_n)^3$ term is negligible, we have that

$$|U^b_{n+1} - U^a_{n+1}| \approx K_2(\Delta t_n)^2.$$

We can then follow the same procedure as before. This scheme (developed by Dogacki and Shampine in 1989) is the basis of the ode23 solver in Matlab.

The first such method to be found was an order five-six pair, discoverd by Fehlberg (working for NASA) in the late sixties. A similar method,

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

based on fourth and fifth order methods, was developed by Dormand and Price (1980), and this is the basis for the ode45 solver in matlab. It's Butcher table is:

0	1						
1/5	1/5						
3/10	3/40	9/40					
4/5	44/45	−56/15	32/9				
8/9	19372/6561	−25360/2187	64448/6561	−212/729			
1	9017/3168	−355/33	46732/5247	49/176	−5103/18656		
1	35/384	0	500/1113	125/192	−2187/6784	11/84	
c_r	35/384	0	500/1113	125/192	−2187/6784	11/84	0
d_r	5179/57600	0	7571/16695	393/640	−92097/339200	187/2100	1/40

Symplectic methods

Sometimes, however, it's not (only) the error that we're interested in.

A very important field of study is dynamical systems, in particular *Hamiltonian systems*. A Hamiltonian system consists of $l = 2m$ differential equations:

$$x_i' = \frac{\partial H}{\partial u_i}$$
$$u_i' = -\frac{\partial H}{\partial x_i}$$

for $i = 1, \ldots, l$. The scalar function $H(\mathbf{x}, \mathbf{u})$ is called the *Hamiltonian*. (see B7.1). A typical example is a system of particles; in this case $\mathbf{x}(t)$ are the positions of particles at time t, and $\mathbf{u}(t)$ are their velocities. In this case H is the total energy.

Example

Consider a simple harmonic oscillator. Let x be the postition at time t, and u be it's velocity. The (scaled) system can be written as

$$x'' = -x \Rightarrow x' = u,\ u' = -x.$$

This can be set in Hamiltonian framework by writing the Hamiltonian

$$H = \frac{1}{2}(x^2 + u^2).$$

Note that the Hamiltonian corresponds to the total energy in the system, and so conservation of energy tells us that solution of the dynamical system correspond to the Hamiltonian being constant – in this case, circles in the (x, u) plane.

Let the solution be given by $\mathbf{u} = (x, u)$. Then the Hamiltonian can be written as $H = \frac{1}{2}\mathbf{u}^T\mathbf{u}$. A reasonable question to ask is how this evolves using the numerical scheme.

Explicit Euler for this system looks like

$$\begin{bmatrix} X_{n+1} \\ U_{n+1} \end{bmatrix} = \begin{bmatrix} X_n \\ U_n \end{bmatrix} + h \begin{bmatrix} U_n \\ -X_n \end{bmatrix} = \begin{bmatrix} 1 & h \\ -h & 1 \end{bmatrix} \begin{bmatrix} U_n \\ X_n \end{bmatrix}$$

and so

$$\begin{bmatrix} X_{n+1} & U_{n+1} \end{bmatrix} \begin{bmatrix} X_{n+1} \\ U_{n+1} \end{bmatrix} = \begin{bmatrix} U_n & X_n \end{bmatrix} \begin{bmatrix} 1 & -h \\ h & 1 \end{bmatrix} \begin{bmatrix} 1 & h \\ -h & 1 \end{bmatrix} \begin{bmatrix} U_n \\ X_n \end{bmatrix} = (1+h^2) \begin{bmatrix} X_n & U_n \end{bmatrix} \begin{bmatrix} X_n \\ U_n \end{bmatrix}$$

and so

$$H_{n+1} = (1 + h^2)H_n.$$

So the Hamiltonian (and hence the energy) grows in time for all time steps.

Now, consider the hybrid Euler scheme

$$\begin{bmatrix} U_{n+1} \\ X_{n+1} \end{bmatrix} = \begin{bmatrix} U_n \\ X_n \end{bmatrix} + h \begin{bmatrix} -X_n \\ U_{n+1} \end{bmatrix}$$

or, in matrix terms:

$$\begin{bmatrix} 1 & -h \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_{n+1} \\ U_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -h & 1 \end{bmatrix} \begin{bmatrix} X_n \\ U_n \end{bmatrix}$$

This also doesn't leave the Hamiltonian H unaltered, but it does preserve a modified Hamiltonian:

$$\hat{H}(x, u) = \frac{1}{2}(x^2 + u^2) - \frac{1}{2}hxu = H(x, u) - \frac{1}{2}hxu.$$

This is called a **symplectic scheme**; while the original Hamiltonian is not preserved, it is recovered in the limit as $h \to 0$.

A common symplectic scheme is the **Stömer-Verlet** scheme. If the system to be solved is

$$\begin{aligned} x' &= u \\ u' &= f(x), \end{aligned}$$

then the Stömer-Verlet scheme takes the form:

$$\begin{aligned} U_{n+1/2} &= U_n + \frac{1}{2}hf(X_n) \\ X_{n+1} &= X_n + hU_{n+1/2} \\ U_{n+1} &= U_{n+1/2} + \frac{1}{2}hf(X_{n+1}), \end{aligned}$$

which, if we eliminate the intermediate value, gives the scheme

$$X_{n+1} = X_n + hU_n + \frac{1}{2}h^2 f(X_n)$$
$$U_{n+1} = U_n + \frac{1}{2}[f(X_n) + f(X_{n+1})]$$

Consider an area in the (x, u) plane, corresponding to a set of initial conditions. Another property of Hamiltonian mechanics is that, as the state of the system progresses, the area remains constant.

Recall that the area of a parallelogram with corners $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ is given by

$$A = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}$$

Consider again the dynamical system

$$x' = u \qquad u' = -x.$$

Now, suppose again that we had three initial conditions $(X_0^1, U_0^1), (X_0^2, U_0^2), (X_0^3, U_0^3)$, and evolve these using Euler's method

$$X_n^i = X_n^i + hU_n^i, \qquad U_n^i = U_n^i - hX_n^i,$$

with area

$$A_n = \begin{vmatrix} X_n^1 & U_n^1 & 1 \\ X_n^2 & U_n^2 & 1 \\ X_n^3 & U_n^3 & 1 \end{vmatrix}$$

To make the algebra easier, let's choose initial (X_0, Y_0), $(X_0 + A, Y_0)$ and $(X_0, Y_0 + B)$.

Then after one step of Euler these move to

$$(X_0 + hU_0, U_0 - hX_0),$$
$$(X_0 + A + hU_0, U_0 - h(X_0 + A)) \text{ and}$$
$$(X_0 + h(U_0 + B), U_0 + B - hX_0)$$

The area is therefore given by

$$\begin{aligned} A &= \begin{vmatrix} X_0 + hU_0 & U_0 - hX_0 & 1 \\ X_0 + A + hU_0 & U_0 - h(X_0 + A) & 1 \\ X_0 + h(U_0 + B) & U_0 + B - hX_0 & 1 \end{vmatrix} \\ &= \begin{vmatrix} X_0 + hU_0 & U_0 - hX_0 & 1 \\ A & -hA & 0 \\ hB & B & 0 \end{vmatrix} \\ &= (1 + h^2)AB. \end{aligned}$$

Therefore the area increases exponentially as Euler progresses. Symplectic methods preserve the area (see problem sheet 2).

NSDE 1: LECTURE 7

TYRONE REES*

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

For the last two weeks we've been looking at one-step methods. These use information (t_n, U_n) that is the most recent step to update the approximate solution U_{n+1} at $t_{n+1} = t_n + h$. To do this, we (in the case of Runge-Kutta methods) evaluate the function multiple times at points in between t_n and t_{n+1} to obtain a more accurate solution.

This can improve the accuracy, may not be the most efficient method, especially in the case where the function is expensive to evaluate.

Recall that the inpiration for one-step methods was re-writing the ODE over a time step as

$$u(t_{n+1}) - u(t_n) = \int_{t_n}^{t_{n+1}} u'(t)dt = \int_{t_n}^{t_{n+1}} f(t,u(t))dt.$$

Runge-Kutta methods evaluated the function at points in between t_n and t_{n+1} in order to better approximate the integral on the right hand side.

Instead, we could integrate over more than one time step, and get a more accurate numerical method by re-using function evaluations we already have. For example,

$$\int_{t_n}^{t_{n+2}} u'(t)dt = \int_{t_n}^{t_{n+2}} f(t,u(t))dt$$

Using, for example, Simpson's rule, we get

$$u(t_{n+2}) - u(t_n) \approx \frac{2h}{6}\left[f(t_{n+2},u(t_{n+2})) + 4f(t_{n+1},u(t_{n+1})) + f(t_n,u(t_n))\right],$$

which suggests the numerical method

$$U_{n+2} = U_n + \frac{h}{3}\left[f(t_{n+2},U_{n+2}) + 4f(t_{n+1},U_{n+1}) + f(t_n,U_n)\right],$$

We're given U_0, we can use a one-step method to find U_1, and then we can use this numerical scheme to approximate the solution at the other time steps.

General form The general form of a linear multistep method is

$$\sum_{j=0}^{k} \alpha_j U_{n+j} = h \sum_{j=0}^{k} \beta_j f(t_{n+j}, U_{n+j})$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

As before, if $\beta_k = 0$ then the method is *explicit*, and if $\beta_k \neq 0$ then the method is *implicit*. (So Simpson's rule is an implicit 2-step method).

As before, we define:

The truncation error:

$$T_n = \frac{\sum_{j=0}^{k}[\alpha_j u(t_{n+j}) - h\beta_j u'(t_{n+j})]}{h\sum_{j=0}^{k}\beta_j}$$

consistency:

$$\lim_{h\to 0, n\to\infty, nh=t_n-t_0} T_n = 0$$

And a method is **pth order accurate** if:

$$|T_n| \leq Kh^p.$$

Note that

$$u(t_{n+j}) = u(t_n) + (jh)u'(t_n) + \frac{(jh)^2}{2!}u''(t_n) + \dots$$

and also

$$u'(t_{n+j}) = u'(t_n) + (jh)u''(t_n) + \frac{(jh)^2}{2!}u'''(t_n) + \dots$$

Subsituting this into T_n we get

$$T_n = \frac{1}{h\sum_{j=1}^{k}\beta_j}[C_0 u(t_n) + C_1 h u'(t_n) + C_2 h^2 u''(t_n) + \cdots]$$

where

$$\begin{aligned}
C_0 &= \sum_{j=0}^{k}\alpha_j, \\
C_1 &= \sum_{j=1}^{k} j\alpha_j - \sum_{j=0}^{k}\beta_j \\
&\dots \\
C_q &= \sum_{j=1}^{k}\frac{j^q}{q!}\alpha_j - \sum_{j=1}^{k}\frac{j^{q-1}}{(q-1)!}\beta_j
\end{aligned}$$

The method is *consistent* if $\lim T_n = 0$, which is equivalent to requiring that $C_0 = 0$ and $C_1 = 0$.

Furthermore, the method is pth order accurate if and only if

$$C_0 = C_1 = \cdots = C_p = 0 \text{ and } C_{p+1} \neq 0$$

and, in this case,

$$T_n = \frac{C_{p+1}}{\sum_{j=1}^{k} \beta_j} h^p u^{(p+1)}(t_n) + O(h^p)$$

C_{p+1} is called the *error constant*.

Adams methods

A particular class of methods, known as Adams methods, have the form

$$U^{n+k} = U^{n+k-1} + h \sum_{j=0}^{k} \beta_j f(t_{n+j}, U_{n+j})$$

i.e., $\alpha_k = 1, \alpha_{k-1} = -1, \alpha_j = 0, j < k-1$.

If we require an explicit method ($\beta_k = 0$), then we can pick the remaining k coefficients to eliminate as many terms as possible in the Taylor expansion. These methods are called **Adams-Bashforth** methods:

$U_{n+1} = U_n + hf(t_n, U_n)$ 1st order

$U_{n+2} = U_{n+1} + \frac{h}{2}(-f(t_n, U_n) + 3f(t_{n+1}, U_{n+1}))$ 2nd order

$U_{n+3} = U_{n+2} + \frac{h}{12}(5f(t_n, U_n) - 16f(t_{n+1}, U_{n+1}) + 23f(t_{n+2}, U_{n+2}))$ 3rd order

$U_{n+4} = U_{n+3} + \frac{h}{23}(-9f(t_n, U_n) + 37f(t_{n+1}, U_{n+1}) - 59f(t_{n+2}, U_{n+2}) + 55f(t_{n+3}, U_{n+3})$ 4th order

If we allow $\beta_k \neq 0$, then we have one more free parameter, and so can get a method of one order higher than the equivalent Adams-Bashforth method. These methods are called **Adams-Moulton** methods:

$U_{n+1} = U_n + \frac{h}{2}(f(t_n, U_n) + f(t_{n+1}, U_{n+1}))$ 2nd order

$U_{n+2} = U_{n+1} + \frac{h}{12}(-f(t_n, U_n) + 8f(t_{n+1}, U_{n+1}) + 5f(t_{n+3}, U_{n+3}))$

$U_{n+3} = U_{n+2} + \frac{h}{24}(f(t_n, U_n) - 5f(t_{n+1}, U_{n+1}) + 19f(t_{n+2}, U_{n+2}) + 9f(t_{n+3}, U_{n+3}))$

Zero-stability

Suppose we have a general $k-$step method

$$\sum_{j=0}^{k} \alpha_j U_{n+j} = h \sum_{j=0}^{k} \beta_j f(t_{n+j}, U_{n+j})$$

U_0 is given, $U_1, \ldots, U_{n-1}$ have to be computed. Question: how do the errors in $U_1, \ldots, U_{k-1}$ affect the later values?

Definintion A linear $k-$step method for

$$u' = f(t,u),\ u(t_0) = u_0,\ t \in [t_0, T_m]$$

is said to be zero-stable if there exists a constant K such that, for any two sequences $U_0, U_1, \ldots, U_{k-1}$, and $\hat{U}_0, \hat{U}_1, \ldots, \hat{U}_{k-1}$,

$$|U_n - \hat{U}_n| \leq K \max\{|U_0 - \hat{U}_0|, |U_1 - \hat{U}_1|, \ldots, |U_{k-1} - \hat{U}_{k-1}|\}$$

for $t_n \leq T_M$ and as $h \to 0$.

This isn't actually useful for checking zero stability – in practice we reformulate in terms of polynomials:

The first characteristic polynomial is given by

$$\rho(z) = \sum_{j=0}^{k} \alpha_j z^j$$

The second characteristic polynomial is given by

$$\sigma(z) = \sum_{j=0}^{k} \beta_j z^j$$

Theorem (Root condition)

A linear multistep method is zero stable for any ODE

$$u' = f(t,u)$$

where f obeys a Lipschitz condition if and only if all zeros of its first characteristic polynomial lie inside the closed unit disk, with any that lie on the unit circle being simple.

Example

Simpson rule method:

$$\begin{aligned} U_{n+2} - U_n &= \frac{h}{3}(f_{n+2} + 4f_{n+1} + f_n) \\ \rho(z) &= z^2 - 1 \\ z &= \pm 1 \end{aligned}$$

(where $f_n = f(t_n, U_n)$).

Simple roots on the unit circle, so the method is zero-stable.

Example

Adams-Bashforth method

$$U_{n+4} - U_{n+3} = \frac{h}{24}(-9f_n + 37f_{n+1} - 59f_{n+2} + 55f_{n+3})$$

$$\rho(z) = z^4 - z^3 = z^3(z-1)$$
$$z_1 = z_2 = z_3 = 0, z_4 = 1$$

Example A 3-step 6-th order accurate method:

$$11U_{n+3} + 27U_{n+2} - 27U_{n+1} - 11U_n = 3h(f_{n+3} + 9f_{n+2} + 9f_{n+1} + f_n)$$

$$\rho(z) = 11z^3 + 27z^2 - 27z - 11$$

$$z_1 = 1, z_2 \approx -0.3189, z_3 \approx -3.1356$$

since $|z_3| > 1$, the method is not zero stable.

NSDE 1: LECTURE 8

TYRONE REES*

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

The general form of a linear multistep method is

$$\sum_{j=0}^{k} \alpha_j U_{n+j} = h \sum_{j=0}^{k} \beta_j f(t_{n+j}, U_{n+j})$$

A method is zero-stable if there exists a constant K such that, for any two sequences $U_0, U_1, \ldots, U_{k-1}$, and $\hat{U}_0, \hat{U}_1, \ldots, \hat{U}_{k-1}$,

$$|U_n - \hat{U}_n| \leq K \max\{|U_0 - \hat{U}_0|, |U_1 - \hat{U}_1|, \ldots, |U_{k-1} - \hat{U}_{k-1}|\}$$

for $t_n \leq T_M$ and as $h \to 0$.

The first characteristic polynomial is given by

$$\rho(z) = \sum_{j=0}^{k} \alpha_j z^j$$

The second characteristic polynomial is given by

$$\sigma(z) = \sum_{j=0}^{k} \beta_j z^j$$

Theorem (Root condition)

A linear multistep method is zero stable for any ODE

$$u' = f(t,u)$$

where f obeys a Lipschitz condition if and only if all zeros of its first characteristic polynomial lie inside the closed unit disk, with any that lie on the unit circle being simple.

Lemma

Consider the kth order homogeneous linear recurrence relation

$$\alpha_k y_{n+k} + \cdots + \alpha_1 y_{n+1} + \alpha_0 y_n = 0, \qquad n = 0, 1, 2, \ldots$$

with $\alpha_k \neq 0$, $\alpha_0 \neq 0$, $\alpha_j \in \mathbb{R}$, $j = 1, \ldots, k$, and the corresponding characteristic polynomial:

$$\rho(z) = \alpha_k z^k + \cdots + \alpha_1 z + \alpha_0.$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

Let z_r, $1 \le r \le l$, $l \le k$, be distinct roots of the polynomial ρ, and let $m_r \ge 1$ denote the multiplicity of z_r, with $m_1 + \cdots + m_l = k$.

If a sequence (y_n) of complex numbers statisfies the recurrence relation above, then

$$y_n = \sum_{r=1}^{l} p_r(n) z_r^n \qquad \forall n \ge 0,$$

where $p_r(\cdot)$ is a polynomial in n of degree $m_r - 1$, $1 \le r \le l$. In particular, if all roots are simple, (i.e. $m_r = 1$, $1 \le r \le k$), then the p_r are constants.

(See, e.g, Suli and Mayers (Lemma 12.1) for a sketch of the proof)

Proof of Root condition (necessity)

We want to prove that zero-stability implies the root condition. Suppose that

$$\sum_{j=0}^{k} \alpha_j U_{n+j} = h \sum_{j=0}^{k} \beta_j f(t_{n+j}, U_{n+j})$$

is zero stable. Then applying the method to the ODE $u' = 0, u(0) = 0$ gives

$$\alpha_k U_{n+k} + \alpha_{k-1} U_{n+k-1} + \cdots + \alpha_1 U_{n+1} + \alpha_o U_n = 0$$

This is a difference equation, and from the lemma, it's general solution is of the form

$$U_n = \sum_s p_s(n) z_s^n,$$

where z_s is a zero of

$$\rho(z) = \sum_{j=0}^{k} \alpha_j z^j$$

of multiplicity $m_s \ge 1$ and p_s is a polynomial of degree $m_s - 1$.

If $|z_s| > 1$ for some s then there are starting values such that the solution grows like $|z_s|^n$.

If $|z_s| = 1$ and z_s has multiplicity $m_s > 1$, then there are starting values such that the solution grows like $n^{m_s - 1}$. In either case, there are solutions which grow unbounded as $n \to \infty$, $h \to 0$, nh fixed.

Consider starting data $U_0, U_1, \ldots, U_{k-1}$ that gives such an unbounded solution, and the starting data

$$\hat{U}_0, \hat{U}_1 = \cdots = \hat{U}_{k-1} = 0,$$

which gives $\hat{U}_n = 0$ for all $n \geq 0$.

Therefore, if a method violates the root condition, it cannot be zero stable.

The proof of the converse is technical, and outside the scope of this course.

Example

Simpson rule method:

$$U_{n+2} - U_n = \frac{h}{3}(f_{n+2} + 4f_{n+1} + f_n)$$
$$\rho(z) = z^2 - 1$$
$$z = \pm 1$$

(where $f_n = f(t_n, U_n)$).

Simple roots on the unit circle, so the method is zero-stable.

Example

Adams-Bashforth method

$$U_{n+4} - U_{n+3} = \frac{h}{24}(-9f_n + 37f_{n+1} - 59f_{n+2} + 55f_{n+3})$$

$$\rho(z) = z^4 - z^3 = z^3(z-1)$$
$$z_1 = z_2 = z_3 = 0, z_4 = 1$$

Example A 3-step 6-th order accurate method:

$$11U_{n+3} + 27U_{n+2} - 27U_{n+1} - 11U_n = 3h(f_{n+3} + 9f_{n+2} + 9f_{n+1} + f_n)$$

$$\rho(z) = 11z^3 + 27z^2 - 27z - 11$$

$$z_1 = 1, z_2 \approx -0.3189, z_3 \approx -3.1356$$

since $|z_3| > 1$, the method is not zero stable.

Convergence

The linear multistep method is said to be *convergent* if, for all initial value problems $u' = f(t,u),\ u(t_0) = u_0$ (which satisfies the assumptions of Picard's theorem),

$$\lim_{h \to 0,\ nh = t - t_0} U_n = u(t)$$

for all $t \in [t_0, T_M]$ and for all solutions $\{U_n\}_{n=0}^N$ with consistent starting condition, i.e., with starting values

$$U_s = \eta_s(h)$$

for which $\lim_{h\to 0} \eta_s(h) = U_0$, $s = 0, 1, \ldots, k-1$.

Convergence is a vital property of a numerical method. It tells us that, if we take smaller and smaller time steps, we will get better and better approximations to the true solution.

NSDE 1: LECTURE 9

TYRONE REES*

$$u' = f(t,u), \qquad u(t_0) = u_0.$$

The general form of a linear multistep method is

$$\sum_{j=0}^{k} \alpha_j U_{n+j} = h \sum_{j=0}^{k} \beta_j f(t_{n+j}, U_{n+j})$$

The linear multistep method is said to be *convergent* if, for all initial value problems $u' = f(t,u),\ u(t_0) = u_0$ (which satisfies the assumptions of Picard's theorem),

$$\lim_{h\to 0,\ nh=t-t_0} U_n = u(t)$$

for all $t \in [t_0, T_M]$ and for all solutions $\{U_n\}_{n=0}^{N}$ with consistent starting condition, i.e., with starting values

$$U_s = \eta_s(h)$$

for which $\lim_{h\to 0} \eta_s(h) = U_0,\ s = 0, 1, \ldots, k-1$.

Convergence is a vital property of a numerical method. It tells us that, if we take smaller and smaller time steps, we will get better and better approximations to the true solution.

Theorem Zero stability is a necessary condition for convergence.

Proof Since the scheme converges for all f, choose $f = 0$. In particular, let $u(0) = 0$ with solution $u = 0$.

Then

$$\sum_{j=0}^{k} \alpha_j U_{n+j} = 0 \tag{0.1}$$

As the method is convergent, $U_n \to 0$ as $h \to 0$, $nh \to t$ for consistent starting values $U_0, \ldots, U_{k-1}$.

Now, let $z = re^{i\theta}$ be a root of $\rho(z) = 0$.

Choose the starting data $U_m = hr^m cos(m\theta)$, which is consistent with the initial condition. Note that U_m satisfies (0.1), since by writing

$$U_m = Re(hr^m e^{im\theta}) = hRe(z^m),$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

then

$$\sum_{j=0}^{k} \alpha_j U_{n+j} = h \sum_{j=0}^{k} Re(z^{m+k}) = h \sum_{j=0} Re(z^n \rho(z)) = 0,$$

since z was a root of $\rho(z) = 0$.

Therefore, the values $U_n = hr^n \cos(n\theta)$ are solutions of the discrete method starting from a consistent set of initial values.

- Suppose $\theta \neq 0, \pi$. then

  $$U_n^2 - U_{n+1}U_{n-1} = h^2 r^{2n} \left[\cos^2(n\theta) - \cos((n+1)\theta)\cos((n-1)\theta)\right]$$

  so

  $$U_n^2 - U_{n+1}U_{n-1} = h^2 r^{2n} \sin^2\theta$$

  The LHS tends to zero as $n \to \infty$, $nh \to t$, as all values converge to zero. The right hand side must therefore also tend to zero. However, this is only possibly if $r \leq 1$.
- if $\theta = 0$ or π, then since

  $$U_n = hr^n \qquad (\theta = 0)$$

  or

  $$U_n = hr^n(-1)^n \qquad (\theta = \pi)$$

  we again have that since $U_n \to 0$, $r \leq 1$.

Now, if z is a double root of $\rho(z)$, $U_n = hnr^n \cos(n\theta)$ also satisfies (0.1), and so $r < 1$. Therefore all roots on the unit circle must be simple.

Hence, if the scheme is convergent, the roots of the first characteristic polynomial satisfy root condition, and hence the scheme is zero-stable.

Theorem

Consistency is a necessary condition for convergence.

Proof

Recall that a linear multistep method is consistent if and only if

$$\sum_{j=0}^{k} \alpha_j = 0, and \sum_{j=0}^{k} j\alpha_j = \sum_{j=0}^{k} \beta_j.$$

Assume that a linear multi-step method is convergent for all functions f.

- Consider the function $f = 0$ with initial value $u(0) = 1$, so the solution is $u = 1$. Now $U_0 = U_1 = \cdots U_{k-1}$ is a consistent set of

initial data, and convergence gives $U_n \to 1$ as $nh \to t$, $h \to 0$. The method is

$$\sum_{j=0}^{k} U_{n+j} = 0,$$

and since, in the limit, $U_{n+s} \to 1$ for $s = 1, \ldots k$, we have $\sum_{j=0}^{k} \alpha_k = 0$, as required.

- Next consider $f = 1$ with initial value $u(0) = 0$, so that $u' = 1$. Here the solution is $u(t) = t$, and so $U_n \to t$ as $nh \to t$, $h \to 0$. The method is

$$\sum_{j=0}^{k} \alpha_j U_{n+j} = h \sum_{j=0}^{k} \beta_j.$$

Convergence tells us that $U_{n+r} = (n + r)h$, so

$$\sum_{j=0}^{k} \alpha_j (n + j)h = h \sum_{j=0}^{k} \beta_j$$

i.e.

$$\sum_{j=0}^{k} \alpha_j n + \sum_{j=1}^{k} j\alpha_j = h \sum_{j=0}^{k} \beta_j$$

since the first term is zero by part (i), we must have

$$\sum_{j=1}^{k} j\alpha_j = h \sum_{j=0}^{k} \beta_j$$

Putting these two results together, convergence implies the scheme will be consistent.

Dahlquist Theorem

For a linear multi-step method that is consistent with the ODE $u' = f(t, u)$, where f obeys a Lipschitz condition, and starting with consistent initial data, zero-stability is necessary and sufficient for convergence.

—

Dahlquist's theorem tells us that if a linear multistep method is not zero-stable, then it's global error cannot be made arbirarily small by taking h sufficiently small. In fact, if the root condition is violated, then no matter how good the initial data is, there exists a solution that will grow by an arbitrarily large number.

NSDE 1: LECTURE 10

TYRONE REES*

Absolute stability

Zero-stability tells us that a method will converge if we take h small enough, but tells us nothing about what the solution will look like for a fixed h.

To motivate us, consider the model problem

$$u' = \lambda u,$$

which has exact solution $u = e^{\lambda t}$. If $Re(\lambda) < 0$, then the solutions decay as t increases.

`euler_stability(0.01), euler_stability(0.05), euler_stability(0.1), euler_stability(0.11), euler_stability(0.15)`

Suppose we apply Euler's method:

$$U_{n+1} = U_n + h\lambda U_n = (1 + h\lambda)U_n.$$

if $|1 + h\lambda| > 1$, the solution will grow at each time step. Note that only the combination $\bar{h} = h\lambda$ matters. The method is stable when $|1+\bar{h}| < 1$. In the complex plane this is equivalent to the unit circle, centered on -1.

To make this more general, consider again the model problem above. Then a linear multistep method for this problem looks like:

$$\sum_{j=0}^{k} (\alpha_j - \lambda h \beta_j) U_{n+j} = 0.$$

The difference equation

$$\sum_{j=0}^{k} (\alpha_j - \bar{h}\beta_j) U_{n+j} = 0$$

has general solution

$$U_n = \sum_s p_s(n) z_s^n,$$

where z_s is a zero of the *stability polynomial*

$$\pi(z, \bar{h}) = \rho(h) - \bar{h}\sigma(z) = \sum_{j=0}^{k} (\alpha_j - \bar{h}\beta_j) z^j$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

Since $\lim_{t\to\infty} u(t) = 0$, we want $\lim_{t\to\infty} U_n = 0$, and so we need $|z_s| < 1$ for all $s = 1, 2, \ldots, k$.

A linear multistep method is called *absolutely stable* in an open set $\mathcal{R}_A$ of the complex plane if, for all $\bar{h} \in \mathcal{R}_A$, all roots $z_s = z_s(\bar{h}), s = 1, \ldots k$ of $\pi(z, \bar{h})$ satisfy $|z_a| < h$.

The set $\mathcal{R}_A$ is called the *region of absolute stability.*

Example: implicit Euler

$$U_{n+1} - U_n = hf(t_{n+1}, U_{n+1}$$

$$\rho(z) = z - 1, \qquad \sigma(z) = z$$

$$\pi(z, \bar{h}) = z - 1 - \bar{h}z = 0$$
$$\iff z = \frac{1}{1 - \bar{h}}$$

$|z| < 1$ when $|1 - \bar{h}| > 1$. Therefore $\mathcal{R}_A = \{\bar{h} \in \mathbb{C} \ : \ |1 - \bar{h}| > 1\}$ This is the outside of the unit circle, centered at 1, in the complex plane. Note that since this contains the whole of the left half plane, the method is stable for all λ where $Re(\lambda) < 0$.

Definition A linear multistep method is said to be A-stable it its region of absolute stability, $\mathcal{R}_A$, contains the whole of the open left-hand complex half plane, $Re(\bar{h}) < 0$.

The implicit Euler method is A-stable.

Dahlquist Barrier Theorem

- No explicit linear multistep method is A-stable
- The order of an A-stable implicit linear multistep method is ≤ 2.
- The second-order A-stable linear multistep method with the smallest error constant is the trapezium rule method.

Stiffness

When deal with systems of differential equations, it is common that the methods we have studied may not work well with some systems where different parts of the solution evolve on different time scales.

Consider

$$u'' + (1 + a)u' + au = 0$$

which we can write as

$$u' = v$$
$$v' = -(1 + a)v - au$$

or, in matrix form

$$\frac{d}{dt}\begin{bmatrix} u \\ v \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ -a & -(1+a) \end{bmatrix}}_{A} \begin{bmatrix} u \\ v \end{bmatrix}$$

which has solutions $u = c_1 e^{-t} + c_2 e^{-at}$.

If $a \gg 1$, then u and v will have different scales, $O(1)$ and $O(a^{-1})$.

Note that the eigenpairs of A such that $A\mathbf{v}_i = \lambda_i \mathbf{v}_i$ are given by

$$\lambda_i = -1, \mathbf{v}_1 = \begin{bmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix}) \text{ and } \lambda_2 = -a, \mathbf{v}_2 = \begin{bmatrix} -1/\sqrt{1+a^2} \\ 1/\sqrt{1+a^2} \end{bmatrix}.$$

Therefore, letting $V = [\mathbf{v}_1, \mathbf{v}_2]$ and $\Lambda = diag(\lambda_1, \lambda_2)$, we can write $A = V\Lambda V^{-1}$

$$\begin{aligned} \mathbf{u}' &= A\mathbf{u} \\ \Rightarrow \mathbf{u}' &= V\Lambda V^{-1}\mathbf{u} \\ \Rightarrow V^{-1}\mathbf{u}' &= \Lambda V^{-1}\mathbf{u} \end{aligned}$$

Making the change of variables

$$\begin{bmatrix} p \\ q \end{bmatrix} = V^{-1} \begin{bmatrix} u \\ v \end{bmatrix}$$

therefore gives the decoupled system of equations

$$\begin{aligned} \frac{dp}{dt} &= -p \\ \frac{dq}{dt} &= -aq. \end{aligned}$$

Treating these separately would lead to very different requirements for stability: the equation for p could take much shorter time steps and still be stable. However, when we solve the systems together, for stability we need to take the time step required for stability of the bottom equation.

Such a system is called *Stiff*.

Another example of a stiff system is Van-der-pol's oscillator:

$$u'' + \mu(u^2 - 1)u' + u = 0$$

`vdp_stiff(0.16), vdp_stiff(0.165), vdp_stiff(0.17)`

NSDE 1: LECTURE 11

TYRONE REES*

The numerical solution of parabolic problems

For the rest of the course we're going to look at problems of the form:

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(a(x,t)\frac{\partial u}{\partial x}\right) + f(x,t),$$

with $u(x,0) = u_0(x)$ and appropriate boundary condititions.

In practice, we're going to concentrate on the model problem:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial t^2}.$$

Before we look at how to solve this numerically, let's explore what the analytic solution looks like. To do this, we're going to use the Fourier transform:

$$\hat{u}(\xi) = F[u](\xi) = \int_{-\infty}^{\infty} u(x)e^{-ix\xi}dx$$

Applying this to the PDE, we get:

$$\begin{aligned}\int_{-\infty}^{\infty} \frac{\partial u}{\partial t}(x,t)e^{-ix\xi}dx &= \int_{-\infty}^{\infty} \frac{\partial^2 u}{\partial t^2}(x,t)e^{-ix\xi}dx \\ \frac{d}{dt}\underbrace{\int_{-\infty}^{\infty} u(x,t)e^{-ix\xi}dx}_{\hat{u}(\xi,t)} &= (i\xi)^2 \underbrace{\int_{-\infty}^{\infty} u(x,t)e^{-ix\xi}dx}_{\hat{u}(\xi,t)}\end{aligned}$$

where we've integrated by parts twice on the RHS and ignored boundary terms at $\pm\infty$. So

$$\frac{d\hat{u}}{dt} = -\xi^2\hat{u}.$$

This has solutions

$$\hat{u}(\xi,t) = A(\xi)e^{-\xi^2 t} = \hat{u}(\xi,0)e^{-\xi^2 t}.$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

Now we just need to apply the inverse Fourier transform:

$$\begin{aligned} u(x,t) &= F^{-1}\left(e^{-\xi^2 t}\hat{u}_0\right) \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{u}_0(\xi) e^{-\xi^2 t} e^{i\xi x} d\xi \\ &= \ldots \text{ messy calculation } \ldots \\ &= \frac{1}{\sqrt{4\pi t}}\int_{-\infty}^{\infty} e^{-(x-y)^2/(4t)} u_0(y) dy \end{aligned}$$

`AnalyticSolution.m`

This behaviour can be explained by **Parseval's identity:**

$$\|u\|^2_{L_2(-\infty,\infty)} = \frac{1}{2\pi}\|\hat{u}\|^2_{L_2(-\infty,\infty)},$$

where

$$\|u\|_{L_2(\infty,\infty)} = \left(\int_{-\infty}^{\infty} |u(x)|^2 dx\right)^{1/2}$$

If we apply this to our function:

$$\begin{aligned} \|u(\cdot,t)\|^2_{L_2(-\infty,\infty)} &= \frac{1}{2\pi}\|\hat{u}(\cdot,t)\|^2_{L_2(-\infty,\infty)} \\ &= \frac{1}{2\pi}\|e^{-\xi^2 t}\hat{u}_0(\cdot)\|^2_{L_2(-\infty,\infty)} \\ &\leq \frac{1}{2\pi}\max_{\xi}|e^{-\xi^2 t}|\|\hat{u}_0\|_2^2 \\ &\leq \frac{1}{2\pi}\|\hat{u}_0\|_2^2 \\ &= \|u_0\|_2^2 \qquad \text{[by Parseval's identity]} \end{aligned}$$

This is an important property of the solution, that we must make sure is satisfied by any numerical approximation.

Proof of Parseval's identity

Let $v(x)$ and $w(x)$ be two functions.

$$\begin{aligned} \int_{-\infty}^{\infty} \hat{w}(x)v(x)dx &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} w(\xi)e^{-i\xi x}d\xi v(x)dx \\ &= \int_{-\infty}^{\infty} w(\xi)\int_{-\infty}^{\infty} e^{-i\xi x}v(x)dxd\xi \\ &= \int_{-\infty}^{\infty} w(\xi)\hat{v}(\xi)d\xi \end{aligned}$$

Now, if we let $w(\xi) = \overline{\hat{v}(\xi)}$, then

$$\hat{w}(\xi) = \int_{-\infty}^{\infty} \overline{\hat{v}(x)} e^{-i\xi x} dx = \overline{\int_{-\infty}^{\infty} \hat{v}(x) e^{i\xi x} dx} = 2\pi \bar{v}$$

Where we've used the fact that $v = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{v} e^{i\xi x} dx$ (by definition of the inverse Fourier transform). Therefore

$$2\pi \int_{-\infty}^{\infty} \bar{v}(x) v(x) dx = \int_{-\infty}^{\infty} \overline{\hat{v}(\xi)} \hat{v}(\xi) ds$$
$$\Rightarrow \|v\|_2^2 = \frac{1}{2\pi} \|\hat{v}\|_2^2$$

Discretization Suppose $x \in [0, X]$ and $t \in [0, T]$, where $T > 0$ is a given final time, and 0, X are the left and right boundaries.

Construct a finite-difference grid:

$$\Delta x = x_{\text{end}}/N \text{ in the } x\text{-direction}$$
$$\Delta t = T/M \text{ in the } t\text{-direction}$$

so that

$$x_j = j\Delta x, \text{ and } t_m = m\Delta t.$$

Since

$$\frac{\partial u}{\partial t}(x_j, t_m) = \lim_{\Delta t \to 0} \frac{u(x_j, t_m + \Delta t) - u(x_j, t_m)}{\Delta t}$$

and

$$\frac{\partial^2 u}{\partial x^2}(x_j, t_m) = \lim_{\Delta x \to 0} \frac{u(x_j + \Delta x, t_m) - 2u(x_j, t_m) + u(x_j - \Delta x, t_m)}{(\Delta x)^2}$$

We can approximate $u(x_j, t_m) \approx U_j^m$, so that

$$\frac{\partial u}{\partial t}(x_j, t_m) = \frac{U_j^{m+1} - U_j^m}{\Delta t}$$

and

$$\frac{\partial^2 u}{\partial x^2}(x_j, t_m) = \frac{U_{j-1}^m - 2U_j^m + U_{j+1}^m}{(\Delta x)^2}.$$

This scheme can be illustrated by the stencil:

Putting it together, suppose we want to approximate the PDE

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$$

for $x \in [0, X]$ and $t \in [0, T]$, where $u(\mathbf{x}, 0) = u_0(\mathbf{x})$, and $u(0, t) = u_a(t)$, $u(X, t) = u_b(t)$. Then we get that

$$U_j^0 = u_0(x_j), \qquad j = 0, \pm 1, \pm 2, \cdots$$

and then we can update U at the next time step by

$$U_0^{m+1} = u_a(t_{m+1}), \;\; U_{N+1}^{m+1} = u_b(t_{m+1})$$
$$\frac{U_j^{m+1} - U_j^m}{\Delta t} = \frac{U_{j-1}^m - 2U_j^m + U_{j-1}^m}{(\Delta t)^2}.$$

The final relation can also be written as

$$U_j^{m+1} = U_j^m + \mu(U_{j-1}^m - 2U_j^m + U_{j+1}^m),$$

where $\mu = \frac{\Delta t}{\Delta x^2}$.

NSDE 1: LECTURE 12

TYRONE REES*

Recap To solve

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$$

for $x \in [0, X]$ and $t \in [0, T]$, where $u(x, 0) = u_0(x)$, and $u(0, t) = u_a(t)$, $u(X, t) = u_b(t)$, set up a grid

$$x_j = j\Delta x, \text{ and } t_m = m\Delta t$$

where

$$\Delta x = X/N \text{ in the } x\text{-direction}$$
$$\Delta t = T/M \text{ in the } t\text{-direction.}$$

We approximate the derivatives by $u(x_j, t_m) \approx U_j^m$, so that the derivatives are approximately

$$\frac{\partial u}{\partial t}(x_j, t_m) \approx \frac{U_j^{m+1} - U_j^m}{\Delta t}$$

and

$$\frac{\partial^2 u}{\partial x^2}(x_j, t_m) \approx \frac{U_{j-1}^m - 2U_j^m + U_{j+1}^m}{(\Delta x)^2}.$$

This gives

$$U_0^{m+1} = u_a(t_{m+1}), \;\; U_{N+1}^{m+1} = u_b(t_{m+1})$$
$$U_j^{m+1} = U_j^m + \mu(U_{j-1}^m - 2U_j^m + U_{j+1}^m)$$

where $\mu = \frac{\Delta t}{\Delta x^2}$.

Stability

If U is a function defined on the infinite grid $x_j = j\Delta x$, $j = 0, \pm 1, \pm 2, \ldots$, the semidiscrete Fourier transform of U is defined as

$$\hat{U}(k) = \Delta x \sum_{j=-\infty}^{\infty} U_j \, e^{-ikx_j}, \qquad k \in [-\pi/\Delta x, \pi/\Delta x]$$

Now, since $\hat{U}(k)$ is a continuous function, we can take the regular inverse Fourier transform to obtain

$$U_j = \frac{1}{2\pi} \int_{-\pi/\Delta x}^{\pi/\Delta x} \hat{U}(k) \, e^{ikj\Delta x} \, dk.$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

It can be shown that there's an associated Parseval's Identity:

$$\|U\|_{\ell_2}^2 = \frac{1}{2\pi}\|\hat{U}\|_{L_2}^2,$$

where

$$\|\hat{U}\|_{L_2} = \left(\int_{-\pi/\Delta x}^{\pi/\Delta x} |\hat{U}(k)|^2 dk\right)^{1/2}$$

and

$$\|U\|_{\ell_2} = \left(\Delta x \sum_{j=-\infty}^{\infty} |U_j|^2\right)^{1/2}$$

[see problem sheet]

Definition: We say that a finite difference scheme for the unsteady heat equation is *(practically) stable* in the ℓ_2 norm if

$$\|U^m\|_{\ell_2} \leq \|U^0\|_{\ell_2},\ m = 1, \dots M,$$

where $U^m = \{U_j^m\}$.

Now, recall the Euler scheme:

$$U_j^{m+1} = U_j^m + \mu(U_{j-1}^m - 2U_j^m + U_{j+1}^m)$$

and inserting the inverse Fourier transform we get

$$\begin{aligned}\frac{1}{2\pi}\int_{-\pi/\Delta x}^{\pi/\Delta x} e^{ikj\Delta x}\hat{U}^{m+1}(k)dk =& \frac{1}{2\pi}\int_{-\pi/\Delta x}^{\pi/\Delta x} e^{ikj\Delta x}\hat{U}^m(k)dk + \\ & \mu\left(\frac{1}{2\pi}\int_{-\pi/\Delta x}^{\pi/\Delta x} e^{ikj-1\Delta x}\hat{U}^m(k)dk\right. \\ & + 2\frac{1}{2\pi}\int_{-\pi/\Delta x}^{\pi/\Delta x} e^{ikj\Delta x}\hat{U}^m(k)dk \\ & \left. + \frac{1}{2\pi}\int_{-\pi/\Delta x}^{\pi/\Delta x} e^{ikj+1\Delta x}\hat{U}^m(k)dk\right)\end{aligned}$$

Rearranging, we get

$$\frac{1}{2\pi}\int_{-\pi/\Delta x}^{\pi/\Delta x} e^{ikj\Delta x}\left(\hat{U}^{m+1}(k) - \hat{U}^m(k) - \mu\left(e^{ik\Delta x} - 2 + e^{-ik\Delta x}\right)\hat{U}^m(k)\right)dk = 0,$$

which, in turn means that

$$\hat{U}^{m+1}(k) = \hat{U}^m(k) + \mu(e^{ik\Delta x} - 2 + e^{ik\Delta x})\hat{U}^m(k)$$

for all wave numbers $k \in [-\pi/\Delta x, \pi/\Delta x]$. Therefore

$$\hat{U}^{m+1}(k) = \lambda(k)\hat{U}^m(k)$$

where $\lambda(k) = 1 + \mu(e^{ik\Delta x} - 2 + e^{ik\Delta x})\hat{U}^m(k)$ Now,

$$\begin{aligned}
\|U^{m+1}\|_{\ell_2} &= \frac{1}{\sqrt{2\pi}}\|\hat{U}^{m+1}\|_{L_2} \qquad \text{Parseval} \\
&= \frac{1}{\sqrt{2\pi}}\|\lambda\hat{U}^m\|_{L_2} \\
&\leq \frac{1}{\sqrt{2\pi}}\max_k|\lambda(k)|\|\hat{U}^m\|_{L_2} \\
&= \max_k|\lambda(k)|\|U^m\|_{\ell_2} \qquad \text{Parseval}
\end{aligned}$$

Since we want that

$$\|U^{m+1}\|_{\ell_2} \leq \|U^m\|_{\ell_2}, \qquad m = 0, 1, 2, \ldots, M-1$$

we need that

$$\max_k|\lambda(k)| \leq 1$$

$$\max_k|1 + \mu(e^{ik\Delta x} - 2 + e^{ik\Delta x})| \leq 1$$

Now, since $e^{ik\Delta x} = \cos k\Delta x + i\sin k\Delta x$, and $e^{-ik\Delta x} = \cos k\Delta x - i\sin k\Delta x$, we can write this as

$$\max_k|1 + 2\mu(\cos k\Delta x - 1)| \leq 1$$

$$\max_k|1 - 4\mu\sin^2\left(\frac{k\Delta x}{2}\right)| \leq 1$$

Now, the condition that

$$-1 \leq 1 - 4\mu\sin^2\left(\frac{k\Delta x}{2}\right) \leq 1 \qquad \forall k \in [-\pi/\Delta x, \pi/\Delta x]$$

holds if and only if

$$\mu = \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}.$$

We've proved the following theorem:

Theorem Suppose that U_j^m is the solution of

$$\frac{U_j^{m+1} - U_j^m}{\Delta t} = \frac{U_{j+1}^m - 2U_j^m + U_{j-1}^m}{(\Delta x)^2}, \ j = 1, 2, \ldots$$

where $U_j^0 = u_0(x_j)$ and $\mu = \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$. Then

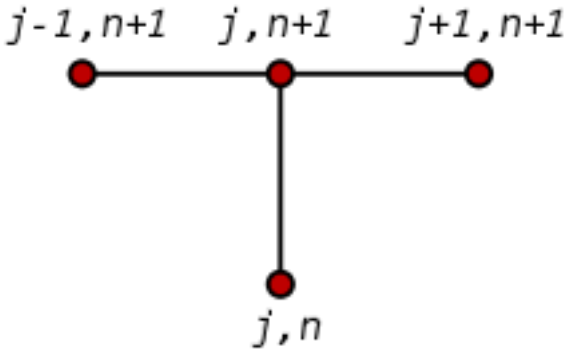
$$\|U^m\|_{\ell_2} \leq \|U^0\|_{\ell_2},\ m = 1, 2, \ldots M.$$

We say that the explicit Euler scheme is *conditionally stable*.

The implicit Euler scheme Consider instead the scheme

$$\frac{U_j^{m+1} - U_j^m}{\Delta t} = \frac{U_{j+1}^{m+1} - 2U_j^{m+1} + U_{j-1}^{m+1}}{(\Delta x)^2},\ j = 1, 2, \ldots$$

where $U_j^0 = u_0(x_j)$



Now we have that

$$U_j^{m+1} - \mu(U_{j+1}^{m+1} - 2U_j^{m+1} + U_{j-1}^{m+1}) = U_j^m,$$

$U_j^0 = u_0(x_j)$, where again $\mu = \Delta t/(\Delta x)^2$).

Using an identical argument to that used for Explicit Euler, we find the amplification factor is here

$$\lambda(k) = \frac{1}{1 + 4\mu \sin^2\left(\frac{k\Delta x}{2}\right)}.$$

In contrast to earlier, this satisfies $|\lambda(k)| \leq 1$ for all values of μ.

Theorem Suppose that U_j^m is the solution of

$$\frac{U_j^{m+1} - U_j^m}{\Delta t} = \frac{U_{j+1}^{m+1} - 2U_j^{m+1} + U_{j-1}^{m+1}}{(\Delta x)^2},\ j = 1, 2, \ldots$$

where $U_j^0 = u_0(x_j)$ and $\mu = \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$. Then

$$\|U^m\|_{\ell_2} \leq \|U^0\|_{\ell_2},\ m = 1, 2, \ldots M.$$

We say that the explicit Euler scheme is *unconditionally stable*.

Fourier modes The same results can be reached using a simplier argument, by inserting the *Fourier mode* into the scheme, i.e. setting

$U_j^m = [\lambda(k)]^m e^{ikj\Delta x}$. For example, substituting this into the explicit Euler scheme:

$$U_j^{m+1} = U_j^m + \mu \left(U_{j+1}^m - 2U_j^m + U_{j-1}^m \right)$$

gives

$$\lambda(k) = 1 + \mu(e^{ik\Delta x} - 2 + e^{-ik\Delta x}),$$

and hence the result follows as before.

NSDE 1: LECTURE 13

TYRONE REES*

Recap To solve

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$$

for $x \in (-\infty, \infty)$ and $t \in [0, T]$, where $u(x, 0) = u_0(x)$, and $u \to 0$ as $\to \pm\infty$ Set up a grid:

$$x_j = j\Delta x, \text{ and } t_m = m\Delta t.$$

We've considered two schemes: Explicit Euler in time with central differences in space:

$$U_j^{m+1} = U_j^m + \mu(U_{j-1}^m - 2U_j^m + U_{j+1}^m)$$

and Implicit Euler in time with central differences in space:

$$U_j^{m+1} - \mu(U_{j+1}^{m+1} - 2U_j^{m+1} + U_{j-1}^{m+1}) = U_j^m,$$

where $\mu = \Delta t/(\Delta x)^2$.

We analyzed stability using the semidiscrete Fourier transform:

$$\hat{U}(k) = \Delta x \sum_{j=-\infty}^{\infty} U_j \, e^{-ikx_j}, \qquad k \in [-\pi/\Delta x, \pi/\Delta x],$$

with inverse:

$$U_j = \frac{1}{2\pi} \int_{-\pi/\Delta x}^{\pi/\Delta x} \hat{U}(k) \, e^{ikj\Delta x} \, dk$$

`aliasing.m`

Fourier modes The same stability results can be reached using a simplier argument, by inserting the *Fourier mode* into the scheme, i.e. setting $U_j^m = [\lambda(k)]^m e^{ikj\Delta x}$. You should be able to do either method.

Consider another scheme: the $\theta-$scheme:

$$\frac{U_j^{m+1} - U_j^m}{\Delta t} = (1-\theta)\frac{U_{j+1}^m - 2U_j^m + U_{j-1}^m}{(\Delta x)^2} + \theta\frac{U_{j+1}^{m+1} - 2U_j^{m+1} + U_{j-1}^{m+1}}{(\Delta x)^2}$$

where $\theta \in [0, 1]$ is a parameter:

- $\theta = 0$: Explicit Euler scheme
- $\theta = 1$: Implicit Euler scheme

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

- $\theta = 1/2$: Crank-Nicolson scheme

To analyse stability, note

$$U_j^{m+1} - U_j^m = (1-\theta)\mu(U_{j-1}^m - 2U_j^m + U_{j+1}^m) + \mu\theta(U_{j-1}^{m+1} - 2U_j^{m+1} + U_{j+1}^{m+1})$$

inserting the Fourier mode:

$$\begin{aligned}[\lambda(k)]^{m+1}e^{ikj\Delta x} - [\lambda(k)]^m e^{ikj\Delta x} &= (1-\theta)\mu([\lambda(k)]^m e^{ik(j-1)\Delta x} - 2[\lambda(k)]^m e^{ikj\Delta x} + [\lambda(k)]^m e^{ik(j+1)\Delta x}) \\ &+ (1-\theta)\mu([\lambda(k)]^{m+1}e^{ik(j-1)\Delta x} - 2[\lambda(k)]^{m+1}e^{ikj\Delta x} + [\lambda(k)]^{m+1}e^{ik(j+1)\Delta x})\end{aligned}$$

Dividing through by $[\lambda(k)]^m e^{ikx_j}$ gives:

$$\lambda(k) - 1 = (1-\theta)\mu(e^{-ik\Delta x} - 2 + e^{ik\Delta x}) + \theta\mu\lambda(k)(e^{-ik\Delta x} - 2 + e^{ik\Delta x}).$$

Therefore, by the same arguments as last time:

$$\lambda(k) - 1 = -4(1-\theta)\mu\sin^2\left(\frac{k\Delta x}{2}\right) - 4\theta\mu\lambda(k)\sin^2\left(\frac{k\Delta x}{2}\right)$$

and so

$$\lambda(k) = \frac{1 - 4(1-\theta)\mu\sin^2\left(\frac{k\Delta x}{2}\right)}{1 + 4\theta\mu\sin^2\left(\frac{k\Delta x}{2}\right)}$$

For practical stability, we require that

$$|\lambda(k)| \leq 1 \quad \forall\ k \in \left[\frac{-\pi}{\Delta x}, \frac{\pi}{\Delta x}\right]$$

i.e.

$$-1 \leq \frac{1 - 4(1-\theta)\mu p}{1 + 4\theta\mu p} \leq 1,$$

where we've written $p = \sin^2\left(\frac{k\Delta x}{2}\right)$, which takes values between 0 and 1. We can write this as

$$-1 \leq 1 - \frac{4\mu p}{1 + 4\theta\mu p} \leq 1$$

This is clearly less than one, and monotonic decreasing, so we just have to check what happens at $p = 1$. We get that the inequality is always satisfied of $\theta \leq 1/2$, and otherwise is satisfied if

$$\mu \leq \frac{1}{2(1-2\theta)}.$$

Therefore:

- For $\theta \in [1/2, 1]$: unconditionally stable

- For $\theta \in [0, 1/2[$: stable if $\mu \leq \frac{1}{2(1-2\theta)}$

Boundary conditions

In practice, we'll be solving on a bounded domain, and so we'll need boundary conditions: As well as an initial condition $u(x,0) = u_0(x)$, we need, e.g.,

- Dirichlet boundary conditions: $u(x_0,t) = u_a(t)$, $u(x_{N+1},t) = u_b(t)$
- Neumann boundary conditions: $\frac{\partial u}{\partial t}(x_0,t) = u_a(t)$, $\frac{\partial u}{\partial t}(x_{N+1},t) = u_b(t)$
- Mixed boundary conditions: $\frac{\partial u}{\partial t}(x_0,t) = u_a(t)$, $u(x_{N+1},t) = u_b(t)$

How do we solve the system?

We can write the θ-scheme as

$$U_j^{m+1} - \theta\mu(U_{j+1}^{m+1} - 2U_j^{m+1} + U_{j-1}^{m+1}) = U_j^m + (1-\theta)\mu(U_{j+1}^m - 2U_j^m + U_{j-1}^m)$$

for $j = 1, \dots N$. In the Dirichlet case, we know $U_j^0 = u_0(x_j)$ for all j, and also $U_0^m = u_a(t_m)$, $U_{N+1}^m = u_b(t_m)$, so we can simply plug these in where needed. This can be be written in matrix form:

$$U^{m+1} - \theta\mu \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & 0 & 1 & -2 \end{bmatrix} U^{m+1} = U^m +$$

$$(1-\theta)\mu \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & 0 & 1 & -2 \end{bmatrix} U^m + (1-\theta)\mu \begin{bmatrix} u_a(t_m) \\ 0 \\ \vdots \\ 0 \\ u_b(t_m) \end{bmatrix} + \theta\mu \begin{bmatrix} u_a(t_{m+1}) \\ 0 \\ \vdots \\ 0 \\ u_b(t_{m+1}) \end{bmatrix},$$

or, if we write

$$K = \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & 0 & 1 & -2 \end{bmatrix}, \qquad \mathbf{f} = \begin{bmatrix} (1-\theta)\mu u_a(t_m) + \theta\mu u_a(t_{m+1}) \\ 0 \\ \vdots \\ 0 \\ (1-\theta)\mu u_b(t_m) + \theta\mu u_b(t_{m+1}) \end{bmatrix}$$

then

$$(I - \theta\mu K)U^{m+1} = (I + (1-\theta)\mu K)U^m + \mathbf{f}$$

At each step, a tridiagonal matrix must be solved (by, e.g., the Thomas algorithm – see problem sheet 5).

NSDE 1: LECTURE 14

TYRONE REES*

Recap

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$$

for $x \in [0, X]$ and $t \in [0, T]$ where $u(x, 0) = u_0(x)$, and with mixed boundary conditions $\frac{\partial u}{\partial x}(0, t) = u_a(t)$, $u(X, t) = u_b(t)$ Set up a grid:

$$x_j = j\Delta x, j = 0, \ldots, N + 1 \text{ and } t_m = m\Delta t, m = 1, \ldots, M.$$

$$U_j^{m+1} - \theta\mu(U_{j+1}^{m+1} - 2U_j^{m+1} + U_{j-1}^{m+1}) = U_j^m + (1-\theta)\mu(U_{j+1}^m - 2U_j^m + U_{j-1}^m)$$

for $j = 1, \ldots N$.

Neumann boundary conditions Suppose that we have a Neumann boundary condition at x_0, so $\frac{\partial u}{\partial x}(x_0, t) = u_a(t)$, with a Dirichlet b.c. at x_{N+1}. Now U_0^m is also unknown, so what do we do?

The solution is to introduce a *ficticious point*, U_{-1}^m, and set

$$\frac{U_1^m - U_{-1}^m}{2\Delta x} = u_a(t_m)$$

(We must use central differences, otherwise the order of accuracy of the boundary condition would be less than that of the PDE). Then $U_{-1}^m = U_1^m - 2\Delta x u_a(t_m)$. We can therefore substitute this into our scheme, giving

$$(I - \theta\mu K^N)U^{m+1} = (I + (1-\theta)\mu K^N)U^m + \mathbf{f}^N$$

where

$$K^N = \begin{bmatrix} -2 & 2 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & 0 & 1 & -2 \end{bmatrix}, \qquad \mathbf{f} = \begin{bmatrix} -2(1-\theta)\mu\Delta x u_a(t_m) - 2\theta\mu\Delta x u_a(t_{m+1}) \\ 0 \\ \vdots \\ 0 \\ (1-\theta)\mu u_b(t_m) + \theta\mu u_b(t_{m+1}) \end{bmatrix}$$

Truncation error

Suppose we solve the heat equation with Dirichlet boundary conditions:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \quad x \in [a, b],\ t \in [0, T]$$
$$u(x, 0) = u_0(x)$$
$$u(a, t) = u_a(t), u(b, t) = u_b(t)$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

using Explicit Euler in time and central differences in space

$$\frac{U_j^{m+1} - U_j^m}{\Delta t} = \frac{U_{j-1}^m - 2U_j^m + U_{j-1}^m}{\Delta x^2}$$

$$U_j^0 = u_0(x_j), j = 1, \ldots, N$$

$$U_0^{m+1} = u_a(t_{m+1}), U_{N+1}^{m+1} = u_b(t_{m+1}), m = 1, \ldots, M$$

We define the truncation error of the scheme as

$$T_j^m = \frac{u_j^{m+1} - u_j^m}{\Delta t} - \frac{u_{j-1}^m - 2u_j^m + u_{j-1}^m}{\Delta x^2}$$

where $u_j^m = u(x_j, t_m)$.

Then, expanding using Taylor series, we get

$$u_j^{m+1} = \left[u + \Delta t u_t + \frac{\Delta t^2}{2} u_{tt} + \frac{\Delta t^3}{6} u_{ttt} + \cdots \right]_j^m$$

and so the time derivative gives

$$\begin{aligned} \frac{u_j^{m+1} - u_j^m}{\Delta t} &= \frac{\left[u + \Delta t u_t + \frac{\Delta t^2}{2} u_{tt} + \frac{\Delta t^3}{6} u_{ttt} + \cdots - u \right]_j^m}{\Delta t} \\ &= u_t + \frac{\Delta t}{2} u_{tt} + \cdots \end{aligned}$$

Now, for the central differences approximation:

$$\begin{aligned} u_{j+1}^m &= \left[u + \Delta x u_x + \frac{\Delta x^2}{2} u_{xx} + \frac{\Delta x^3}{6} u_{xxx} + \frac{\Delta x^4}{24} u_{xxxx} + \cdots \right]_j^m \\ -2u_j^m &= -2u_j^m \\ u_{j-1}^m &= \left[u - \Delta x u_x + \frac{\Delta x^2}{2} u_{xx} - \frac{\Delta x^3}{6} u_{xxx} + \frac{\Delta x^4}{24} u_{xxxx} + \cdots \right]_j^m \end{aligned}$$

and so we get

$$\begin{aligned} \frac{u_{j+1}^m - 2u_j^m + u_{j-1}^m}{\Delta x^2} &= \frac{\Delta x^2 u_{xx} + \frac{\Delta x^4}{12} u_{xxxx} + \cdots}{\Delta x^2} \\ &= u_{xx} + \frac{\Delta x^2}{12} u_{xxxx} + \cdots \end{aligned}$$

Therefore, putting it together:

$$T_j^m = u_t - u_{xx} + \frac{\Delta t}{2} u_{tt} - \frac{\Delta x^2}{12} u_{xxxx} + \cdots = O(\Delta t + (\Delta x)^2)$$

For other schemes, choose the point that you expand about accordingly to minimize the algebra!

- Explicit Euler: u_j^m – $O(\Delta t + \Delta x^2)$
- Implicit Euler: u_j^{m+1} – $O(\Delta t + \Delta x^2)$
- Crank-Nicolson: $u_j^{m+1/2}$ – $O(\Delta t^2 + \Delta x^2)$

Error analysis of the Explicit Euler scheme Let us define the *global error* as

$$e_j^m = u(x_j, t_m) - U_j^m.$$

Note that, for a Dirichlet problem,

$$e_0^{m+1} = 0, e_{N+1}^{m+1} = 0, e_j^0 = 0, j = 1, \ldots, N$$

Then, we have

$$\begin{aligned} U_j^{m+1} &= U_j^m + \mu(U_{j-1}^m - 2U_j^m + U_{j+1}^m) \\ u_j^{m+1} &= u_j^m + \mu(u_{j-1}^m - 2u_j^m + u_{j+1}^m) + \Delta t T_j^m \end{aligned}$$

and so

$$\begin{aligned} e_j^{m+1} &= e_j^m + \mu(e_{j-1}^m - 2e_j^m + e_{j+1}^m) + \Delta t T_j^m \\ &= (1 - 2\mu)e_j^m + \mu e_{j-1}^m + \mu e_{j+1}^m + \Delta t T_j^m \end{aligned}$$

Let $E^m = \max_{0 \le j \le N+1} |e_j^m|$ and $T^m = \max_{1 \le j \le N} |T_j^m|$. As long as $(1 - 2\mu) \ge 0$ (recall that this scheme is absolutely stable if $\mu \le 1/2$) we can write

$$\begin{aligned} E^{m+1} &\le (1 - 2\mu)E^m + \mu E^m + \mu E^m + \Delta t T^m \\ &= E^m + \Delta t T^m \end{aligned}$$

Therefore, since $E^0 = 0$,

$$\begin{aligned} E^m &\le \Delta t \sum_{i=0}^{m-1} T^i \\ &\le m\Delta t \max_{0 \le i \le m-1} T^i \\ &\le T \max_{0 \le m \le M} \max_{1 \le j \le N} |T_j^m| \end{aligned}$$

Therefore, since the Euler scheme has truncation error

$$T_j^m = O(\Delta x^2 + \Delta t)$$

we have that

$$\max_{0\leq m\leq M}\max_{1\leq j\leq N}|u(x_j,t_m)-U_j^m|\leq Const.(\Delta x^2+\Delta t)$$

NSDE 1: LECTURE 15

TYRONE REES*

Another concept of Stability

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$$

for $x \in (-\infty, \infty)$ and $t \in [0, T]$, where $u(x, 0) = u_0(x)$, and $u(x, t) \to 0$ as $x \to \pm\infty$.

We shall say that a finite difference scheme for this problem is *von Neumann*-stable in the ℓ_2 norm if there exists a positive constant $C = C(T)$ such that

$$\|U^m\|_{\ell_2} \le C\|U^0\|_{\ell_2}, \qquad m = 1, \ldots, M = T/\Delta t,$$

where, as usual

$$\|U^m\|_{\ell_2} = \left(\Delta x \sum_{j=-\infty}^{\infty} |U_j^m|^2\right)^{1/2}.$$

Note that practical stability $\Rightarrow$ von Neumann stability.

Lemma

If

$$\hat{U}^{m+1}(k) = \lambda(k)\hat{U}^m(k)$$

and

$$|\lambda(k)| \le 1 + C\Delta t \qquad \forall k \in [-\pi/\Delta x, \pi/\Delta x],$$

then the scheme is von Neumann-stable.

Proof

By Parseval's identity for the semidiscrete Fourier Transform:

$$\begin{aligned}
\|U^{m+1}\|_{\ell_2} &= \frac{1}{\sqrt{2\pi}}\|\hat{U}^{m+1}\|_{L_2} \\
&= \frac{1}{\sqrt{2\pi}}\|\lambda(k)\hat{U}^m\|_{L_2} \\
&\le \frac{1}{\sqrt{2\pi}} \max_k |\lambda(k)| \|\hat{U}^m\|_{L_2} \\
&= \max_k |\lambda(k)| \|U^m\|_{\ell_2}
\end{aligned}$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

and hence

$$\|U^{m+1}\|_{\ell_2} \leq (1 + C\Delta t)\|U^m\|_{\ell_2}, \qquad m = 0, 1, \ldots, M-1.$$

Applying this result repeatedly, we get that

$$\begin{aligned}
\|U^m\|_{\ell_2} &\leq (1 + C\Delta t)^m \|U^0\|_{\ell_2}, \qquad m = 0, 1, \ldots, M-1. \\
&\leq (1 + C\Delta t)^M \|U^0\|_{\ell_2} \qquad [\text{as } 1 + C\Delta t > 1, M > m] \\
&\leq e^{C\Delta t M} \|U^0\|_{\ell_2} \qquad [\text{as } 1 + x \leq e^x, x > 0] \\
&= e^{CT} \|U^0\|_{\ell_2} \qquad [\text{as } M = T/\Delta t]
\end{aligned}$$

□

Note that, as is apparent from the proof, von-Neumann stability isn't helpful over long time periods. However, using it we can state an important theorem:

Theorem: The Lax-equivalence theorem For a consistent difference approximation to a well posed linear evolutionary problem, stability as $\Delta t \to 0$ is necessary and sufficient for convergence.

(proof omitted)

This is the PDE equivalent of Dahlquist's theorem for multistep methods for ODEs.

The discrete Maximum principle

Consider the Dirchlet problem. Mathematically, the maximum of the solution must lie on the boundary (either initially, or at a space boundary). It is important that our numerical scheme also satisfies this property.

Proposition: the discrete Maximum principle The θ-method with $0 \leq \theta \leq 1$ and $\mu(1-\theta) \leq 1/2$ gives approximations U_j^m satisfying:

$$U_{min} \leq U_j^m \leq U_{max},$$

where

$$U_{min} = \min \left\{ \min_{0 \leq m \leq M} \{U_0^m\}, \min_{0 \leq j \leq N+1} \{U_j^0\}, \min_{0 \leq m \leq T} \{U_{N+1}^m\} \right\}$$

and

$$U_{max} = \max \left\{ \max_{0 \leq m \leq M} \{U_0^m\}, \max_{0 \leq j \leq N+1} \{U_j^0\}, \max_{0 \leq m \leq T} \{U_{N+1}^m\}. \right\}$$

Proof We can rewrite the $\theta-$scheme as

$$\begin{aligned}
(1 + 2\theta\mu)U_j^{m+1} =& \theta\mu(U_{j+1}^{m+1} + U_{j-1}^{m+1}) \\
&+ (1-\theta)\mu(U_{j+1}^m + U_{j-1}^m) \\
&+ [1 - 2(1-\theta)\mu]U_j^m.
\end{aligned}$$

By hypothesis:

$$\theta\mu \geq 0 \quad (1-\theta)\mu \geq 0 \quad 1-2(1-\theta)\mu \geq 0,$$

with the last of these being from the assumption that $\mu(1-\theta) \leq 1/2$.

Now, suppose that U attains it's maximum at an *internal* grid point U_j^{m+1}, so $1 \leq j \leq N$, $0 \leq m \leq M-1$ (if not, the proof is complete). Define

$$U^\star = \max\{U_{j+1}^{m+1}, U_{j-1}^{m+1}, U_{j+1}^{m}, U_{j-1}^{m}, U_j^m\}.$$

Then

$$\begin{aligned}(1+2\theta\mu)U_j^{m+1} &\leq \theta\mu U^\star + 2(1-\theta)\mu U^\star + [1-2(1-\theta)\mu]U^\star \\ &= (1+2\theta\mu)U^\star\end{aligned}$$

and, so

$$U_j^{m+1} \leq U^\star$$

However, since U_j^{m+1} is assumed to be the maximum value overall

$$U^\star \leq U_j^{m+1},$$

and, therefore,

$$U_j^{m+1} = U^\star.$$

The maximum is therefore also attained at the points neighbouring (x_j, t_{m+1}). The same argument applies to these neighbouring points, and can be continued until the boundary is reached.

Therefore, the maximum is attained at a boundary point.

□

Note that the θ-scheme obeys the discrete maximum principle for

$$\mu(1-\theta) \leq 1/2.$$

This is more demanding than the stability condition:

$$\mu(1-2\theta) \leq 1/2 \qquad \text{for } 0 \leq \theta \leq 1/2.$$

So, e.g., Crank-Nicolson is unconditionally stable, yet it only obeys the maximum principle for

$$\mu = \frac{\Delta t}{\Delta x^2} \leq 1.$$

NSDE 1: LECTURE 16

TYRONE REES*

Let $\Omega = (a,b) \times (c,d)$. Consider the 2D heat equation:

$$\frac{\partial u}{\partial t} = \nabla^2 u \qquad (x,y) \in \Omega,\ t \in (0,T]$$

subject to the initial condition:

$$u(x,y,0) = u_0(x,y), \qquad (x,y) \in \bar{\Omega}$$

and the Dirichlet boundary condition:

$$u|_{\partial\Omega} = B(x,y,t), \qquad t \in [0,T],\ (x,y) \in \partial\Omega,$$

where $\partial\Omega$ is the boundary of Ω.

Define a grid:

$$\Delta x = (b-a)/(N_x+1),\ \Delta y = (d-c)/(N_y+1),\ \Delta t = T/M,$$

and set

$$\begin{aligned} x_i &= a + i\Delta x,\ i = 0,\ldots,N_x+1 \\ y_j &= c + j\Delta y,\ j = 0,\ldots,N_y+1 \\ t_m &= m\Delta t,\ m = 0,\ldots,M. \end{aligned}$$

Let us define

$$\begin{aligned} \delta_x^2 U_{i,j} &= U_{i+1,j} - 2U_{i,j} + U_{i-1,j} \\ \delta_y^2 U_{i,j} &= U_{i,j+1} - 2U_{i,j} + U_{i,j-1} \end{aligned}$$

Then the 2D $\theta-$scheme is given by

$$\frac{U_{i,j}^{m+1} - U_{i,j}^m}{\Delta t} = (1-\theta)\left(\frac{\delta_x^2 U_{i,j}^m}{\Delta x^2} + \frac{\delta_y^2 U_{i,j}^m}{\Delta y^2}\right) + \theta\left(\frac{\delta_x^2 U_{i,j}^{m+1}}{\Delta x^2} + \frac{\delta_y^2 U_{i,j}^{m+1}}{\Delta y^2}\right)$$

$$\begin{aligned} U_{i,j}^0 &= u_0,\ i = 0,\ldots,N_x+1,\ j = 0,\ldots,N_y+1 \\ U_{i,j}^{m+1} &= B(x_i,y_j,t_{m+1}),\ (x_i,y_j) \in \partial\Omega,\ m = 0,\ldots,M-1. \end{aligned}$$

We usually order the vector of unknowns in what's called a Lexicographic ordering:

$$U^m = [U_{1,1}^m,\ldots,U_{N_x,1},U_{1,2},\ldots,U_{N_x,2},\ldots,U_{N_x,Ny}]^T$$

*Rutherford Appleton Laboratory, Chilton, Didcot, UK, tyrone.rees@stfc.ac.uk

If we do this, then the matrix to be solved in an implicit method has 5 diagonals (yet not all next to each other).

Stability

As before, we analyze stability by doing a Fourier analysis, or by inserting the Fourier mode, in this case

$$U_{i,j}^m = [\lambda(k_x, k_y)]^m \, e^{i(k_x x_i + k_y y_j)}$$

This gives

$$\begin{aligned}\lambda - 1 = & -4(1-\theta)\left[\mu_x \sin^2\left(\frac{k_x \Delta x}{2}\right) + \mu_y \sin^2\left(\frac{k_y \Delta y}{2}\right)\right] \\ & - 4\theta\lambda\left[\mu_x \sin^2\left(\frac{k_x \Delta x}{2}\right) + \mu_y \sin^2\left(\frac{k_y \Delta y}{2}\right)\right]\end{aligned}$$

where

$$\mu_x = \frac{\Delta t}{\Delta x^2}, \qquad \mu_y = \frac{\Delta t}{\Delta y^2}$$

and so

$$\lambda = \frac{1 - 4(1-\theta)\left[\mu_x \sin^2\left(\frac{k_x \Delta x}{2}\right) + \mu_y \sin^2\left(\frac{k_y \Delta y}{2}\right)\right]}{1 + 4\theta\left[\mu_x \sin^2\left(\frac{k_x \Delta x}{2}\right) + \mu_y \sin^2\left(\frac{k_y \Delta y}{2}\right)\right]}$$

For practical stability in ℓ_2, we require that

$$|\lambda(k_x, k_y)| \leq 1 \;\forall\; (k_x, k_y) \in \left[-\frac{\pi}{\Delta x}, \frac{\pi}{\Delta x},\right] \times \left[-\frac{\pi}{\Delta y}, \frac{\pi}{\Delta y}\right]$$

which demands

$$-1 \leq \frac{1 - 4(1-\theta)\left[\mu_x + \mu_y\right]}{1 + 4\theta\left[\mu_x + \mu_y\right]} \leq 1,$$

and so

$$2(1-2\theta)(\mu_x + \mu_y) \leq 1$$

Implicit Euler($\theta = 1$)	unconditionally stable
Crank-Nicolson($\theta = 1/2$)	unconditionally stable
Explicit Euler($\theta = 0$)	conditionally stable :

$$\mu_x + \mu_y = \Delta t\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right) \leq \frac{1}{2}$$

Discrete maximum principle
We can write the $\theta-$scheme as

$$\begin{aligned}(1+2\theta(\mu_x+\mu_y))U_{i,j}^{m+1} =&(1-2(1-\theta)(\mu_x+\mu_y))U_{i,j}^{m}\\ &+(1-\theta)\mu_x(U_{i+1,j}^{m}+U_{i-1,j}^{m})\\ &+(1-\theta)\mu_y(U_{i,j+1}^{m}+U_{i,j-1}^{m})\\ &+\theta\mu_x(U_{i+1,j}^{m+1}+U_{i-1,j}^{m+1})\\ &+\theta\mu_y(U_{i,j+1}^{m+1}+U_{i,j-1}^{m+1}) \qquad (*)\end{aligned}$$

If $(\mu_x+\mu_y)(1-\theta)\leq 1/2$, then the $\theta-$scheme obeys a Discrete maximum principle, so that

$$U_{min}\leq U_{i,j}^{m}\leq U_{max},$$

where

$$U_{min}=\min\left\{\min\left\{U_{i,j}^{0}\right\},\min\left\{U_{i,j}^{m}\right\}_{(x_i,y_j)\in\partial\Omega}\right\}$$

and

$$U_{max}=\max\left\{\max\left\{U_{i,j}^{0}\right\},\max\left\{U_{i,j}^{m}\right\}_{(x_i,y_j)\in\partial\Omega}\right\}$$

Proof: exactly similarly to the 1D proof.
Summary For

$$(\mu_x+\mu_y)(1-\theta)\leq 1/2$$

The $\theta-$scheme obeys the discrete maximum principle. This is more demanding than the ℓ_2-stability condition:

$$(\mu_x+\mu_y)(1-2\theta)\leq 1/2,\quad 0\leq\theta\leq 1/2$$

Error analysis
We define the truncation error

$$T_{i,j}^{m}=\frac{u_{i,j}^{m+1}-u_{i,j}^{m}}{\Delta t}-(1-\theta)\left(\frac{\delta_x^2u_{i,j}^{m}}{\Delta x^2}+\frac{\delta_y^2u_{i,j}^{m}}{\Delta y^2}\right)+\theta\left(\frac{\delta_x^2u_{i,j}^{m+1}}{\Delta x^2}+\frac{\delta_y^2u_{i,j}^{m+1}}{\Delta y^2}\right),$$

where $u_{i,j}^{m}=u(x_i,y_j,t_m)$. After applying some Taylor expansions we get

$$T_{i,j}^{m}=\begin{cases}O(\Delta x^2+\Delta y^2+\Delta t^2), & \theta=1/2\\ O(\Delta x^2+\Delta y^2+\Delta t), & \theta\neq 1/2\end{cases}$$

We now look at the global error. First, note that we can rearrange the truncation error formula to give

$$\begin{aligned}(1+2\theta(\mu_x+\mu_y))u_{i,j}^{m+1} =&(1-2(1-\theta)(\mu_x+\mu_y))u_{i,j}^{m}\\ &+(1-\theta)\mu_x(u_{i+1,j}^{m}+u_{i-1,j}^{m})\\ &+(1-\theta)\mu_y(u_{i,j+1}^{m}+u_{i,j-1}^{m})\\ &+\theta\mu_x(u_{i+1,j}^{m+1}+u_{i-1,j}^{m+1})\\ &+\theta\mu_y(u_{i,j+1}^{m+1}+u_{i,j-1}^{m+1})\\ &+\Delta t T_{i,j}^{m}. \qquad (**)\end{aligned}$$

Defining the global error

$$e_{i,j}^{m}=U_{i,j}^{m}-u(x_i,y_j,t_m)$$

then $e_{i,j}^{0}=0$ and $e_{i,j}^{m}=0$ for $(x_i,y_j)\in\partial\Omega$, and, subtracting (*) from (**), we get

$$\begin{aligned}(1+2\theta(\mu_x+\mu_y))e_{i,j}^{m+1} =&(1-2(1-\theta)(\mu_x+\mu_y))e_{i,j}^{m}\\ &+(1-\theta)\mu_x(e_{i+1,j}^{m}+e_{i-1,j}^{m})\\ &+(1-\theta)\mu_y(e_{i,j+1}^{m}+e_{i,j-1}^{m})\\ &+\theta\mu_x(e_{i+1,j}^{m+1}+e_{i-1,j}^{m+1})\\ &+\theta\mu_y(e_{i,j+1}^{m+1}+e_{i,j-1}^{m+1})\\ &+\Delta t T_{i,j}^{m}\end{aligned}$$

Then if $E^m=\max_{i,j}|e_{i,j}^{m}|$ and $T^m=\max_{i,j}|T_{i,j}^{m}|$, and we assume that

$$1-2(1-\theta)(\mu_x+\mu_y)\geq 0,$$

(Discrete Maximum Principle) then we have

$$(1+2\theta(\mu_x+\mu_y))E^{m+1}\leq 2\theta(\mu_x+\mu_y)E^{m+1}+E^m+\Delta t T^m$$

and hence

$$E^{m+1}\leq E^m+\Delta t T^m.$$

As in the 1D case, as $E^0=0$,

$$E^m\leq T\max_m\max_{i,j}|T_{i,j}^{m}|$$

(where T is the maximum time), and so

$$\max_m\max_{i,j}|u(x_i,y_j,t_m)-U_{i,j}^{m}|\leq T\max_m\max_{i,j}|T_{i,j}^{m}|$$