Using the θ-method to solve ODEs

Kathryn Gillow

26th October 2022

1 Introduction

In this report we use the θ-method to solve ODE's. We begin by introducing the method and deriving it's truncation error. We use this to derive an expression for the local error. We then show an example to confirm that the correct rates of convergence are achieve.

In what follows we consider initial value problems of the form

$$\frac{\mathrm{d}u}{dt} = f(t,u) \tag{1}$$

for $t > 0$ with an initial condition $u(0) = u_0$. Here, we assume that f(t,u) satisfies a Lipschitz condition in its second argument and that f(u,t) is bounded.

It is also possible to use the θ-method to solve problems with spacial dependence. For example we could consider the heat equation of the form

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

for $(x,t) \in (-1,1)x(0,T]$ with boundary and initial conditions

$$\begin{aligned} u(x,0) &= u_0(x) \quad for \ -1 < x < 1, \\ u(-1,t) &= g_1(t) \quad for \ t > 0, \\ u(1,t) &= g_2(t) \quad for \ t > 0. \end{aligned}$$

However, we don't consider such problems here. Instead, we refer the interested reader to Ref. [1].

2 The θ-method

For a general introduction to the θ method see Ref. [2]. We summarise the key points here.

In the θ-method we approximate the solution to Equation (1) at a set of discrete time points $t_n = n\Delta t$ for $n = 0, \ldots, N$ where $N \geq 2$ and $N\Delta t = T$, where T is the final time. We let U_n be the numerical approximation to $u(t_n)$.

The θ-method for Equation 1 is

$$\frac{U_{n+1} - U_n}{\Delta t} = \theta f(t_{n+1}, U_n + 1) + (1-\theta) f(t_n, U_n)$$

Where θ is between 0 and 1. We require this equation to hold for $n = 0, \ldots, N-1$ and we apply the initial condition via $U_0 = u_0$. 3 values of θ lead to methods with a specific name:

- $\theta = 0$ is the explicit Euler scheme(also known as "forward Euler")
- $\theta = 1$ is the implicit Euler scheme(also known as "backward Euler")
- $\theta = \frac{1}{2}$ is the Crank Nicolson scheme.

2.0.1 Truncation Error

The truncation error for the θ-method is defined as

$$T_n = \frac{u_{n+1} - u_n}{\Delta t} - \theta f(t_{n+1}, u_{n+1}) - (1-\theta) f(t_n, u_n) \,, \tag{2}$$

where $u_n = u(t_n)$ is the exact solution at the point t_n. The truncation error can be computed using Taylor series expansions about an appropriately chosen time point.

For $\theta = 0$ (i.e. explicit Euler), the expansions are usually performed about $t = t_n$, while for $\theta = 1$ (i.e. implicit Euler), the expansions are usually performed about $t = t_{n+1}$. For general values of θ it is standard to expand about $t_{n+1/2} = (t_n + t_{n+1})/2 = t_n + 1/2\Delta t$.

Note that since $u'(t_n) = f(t_n, u(t_n))$, we may re-write the expression for the truncation error

$$\begin{aligned} T_n &= \frac{u_{n+1} - u_n}{\Delta t} - \theta f(t_{n+1}, u_{n+1}) - (1-\theta) f(t_n, u_n) \\ &= \frac{u_{n+1} - u_n}{\Delta t} - \theta u'(t_{n+1}) - (1-\theta) u'(t_n) \,. \end{aligned} \tag{3}$$

We have

$$\begin{aligned} u(t_n) &= u(t_{n+1/2} - \Delta t/2) \\ &= u(t_{n+1/2}) - \frac{\Delta t}{2} u'(t_{n+1/2}) + \frac{1}{2}\left(\frac{\Delta t}{2}\right)^2 u''(t_{n+1/2}) + \mathcal{O}(\Delta t^3) \end{aligned}$$

Similarly,

$$u(t_{n+1}) = u(t_{n+1/2}) + \frac{\Delta t}{2} u'(t_{n+1/2}) + \frac{1}{2}\left(\frac{\Delta t}{2}\right)^2 u''(t_{n+1/2}) + \mathcal{O}(\Delta t^3)$$

We can also expand the first derivatives in Equation (3):

$$\begin{aligned} u'(t_n) &= u'(t_{n+1/2}) - \frac{\Delta t}{2} u''(t_{n+1/2}) + \mathcal{O}(\Delta t^2) \ , \\ u'(t_{n+1}) &= u'(t_{n+1/2}) + \frac{\Delta t}{2} u''(t_{n+1/2}) + \mathcal{O}(\Delta t^2) \ . \end{aligned}$$

Substituting these four expansions into (3) gives

$$\begin{aligned} T_n = \ & \frac{1}{\Delta t}((u(t_{n+1/2}) + \frac{\Delta t}{2} u'(t_{n+1/2}) + \frac{1}{2}(\frac{\Delta t}{2})^2 u''(t_{n+1/2})) \\ & -(u(t_{n+1/2}) - \frac{\Delta t}{2} u'(t_{n+1/2}) + \frac{1}{2}(\frac{\Delta t}{2})^2 u''(t_{n+1/2}))) \\ & -\theta(u'(t_{n+1/2}) + \frac{\Delta t}{2} u''(t_{n+1/2})) - (1-\theta)(u'(t_{n+1/2}) - \frac{\Delta t}{2} u''(t_{n+1/2})) + O(\Delta t^2) \end{aligned} \quad (4)$$

Many of the terms in (4) cancel so the truncation error simplifies to

$$T_n = \frac{\Delta t}{2}(1-2\theta) u''(t_{n+1/2}) + \mathcal{O}(\Delta t^2) \ .$$

It can be shown by writing out the the $\mathcal{O}(\Delta t^2)$ terms in full, that they do not cancel for any value of θ.

Thus we have shown that for constant θ

$$T_n = \begin{cases} \mathcal{O}(\Delta t) & \text{for } \theta \neq 1/2 \\ \mathcal{O}(\Delta t^2) & \text{for } \theta = 1/2 \end{cases}$$

so that the truncation error of the Crank Nicolson scheme converges twice as fast as that of all other *theta*-methods.

2.1 Pointwise Errors

Recall the definition of the θ-method (**??**) and the corresponding truncation error (2):

$$\begin{aligned} \frac{U_{n+1}-U_n}{\Delta t} &= \theta f(t_{n+1},U_{n+1}) + (1-\theta) f(t_n,U_n) \ , \\ T_n &= \frac{u_{n+1}-u_n}{\Delta t} - \theta f(t_{n+1},u_{n+1}) - (1-\theta) f(t_n,u_n) \ . \end{aligned}$$

We re-arrange both of these to get

$$U_{n+1} = U_n + \Delta t \left(\theta f(t_{n+1},U_{n+1}) + (1-\theta) f(t_n,U_n)\right) \tag{5}$$

$$u_{n+1} = u_n + \Delta t \left(\theta f(t_{n+1},u_{n+1}) + (1-\theta) f(t_n,u_n)\right) + \Delta t T_n \tag{6}$$

$$\begin{aligned} \implies |u_{n+1}-U_{n+1}| &\le |u_n - U_n| + \theta \Delta t |f(t_{n+1},u_{n+1}) - f(t_{n+1},U_{n+1})| \\ &\quad + (1-\theta)\Delta t |f(t_n,u_n) - f(t_n,U_n)| + \Delta t |T_n| \ . \end{aligned} \tag{7}$$

Next suppose that the right-hand-side function $f(t,u)$ satisfies a Lipschitz condition in its second argument, with Lipschitz constant L, so that:

$$|f(t,u) - f(t,v)| \ \le \ L|u-v| \ , \quad \forall (t,u),\ (t,v) \in \Omega \ .$$

We can use this in (7) to get

$$|u_{n+1}-U_{n+1}| \ \le \ |u_n - U_n| + \theta \Delta t L |u_{n+1}-U_{n+1}| + (1-\theta)\Delta t L |u_n - U_n| + \Delta t |T_n| \ .$$

We can re-arrange this to get (for $\Delta t << 1$)

$$\begin{aligned} (1 - L\theta\Delta t)|u_{n+1}-U_{n+1}| &\le (1 + L(1-\theta)\Delta t)|u_n - U_n| + \Delta t |T_n| \\ &\le (1 + L(1-\theta)\Delta t)|u_n - U_n| + \Delta t T_{\max} \ , \end{aligned} \tag{8}$$

where

$$T_{\max} = max_{0 \le n \le N} |T_n|$$

is an upper bound on the absolute value of the truncation error.

Now let $e_n = u_n - U_n$ denote the error at time t_n. Then (8) can be written as

$$|e_{n+1}| \ \le \ \frac{1 + L(1-\theta)\Delta t}{1 - L\theta\Delta t} |e_n| + \frac{\Delta t T_{\max}}{1 - L\theta\Delta t} \ . \tag{9}$$

We can show by induction that

$$\begin{aligned} |e_n| &\leq \left(\frac{1+L(1-\theta)\Delta t}{1-L\theta\Delta t}\right)^n |e_0| + \frac{\Delta t T_{\max}}{1-L\theta\Delta t}\sum_{r=1}^{n}\left(\frac{1+L(1-\theta)\Delta t}{1-L\theta\Delta t}\right)^{r-1} \\ &\leq \left(\frac{1+L(1-\theta)\Delta t}{1-L\theta\Delta t}\right)^n |e_0| + \frac{T_{\max}}{L}\left[\left(\frac{1+L(1-\theta)\Delta t}{1-L\theta\Delta t}\right)^n - 1\right] , \end{aligned}$$

where the final line comes from evaluating the sum and simplifying. This holds for $n = 0, 1, \ldots, N$.

In practice, we usually set $U_0 = u_0$ which means that $e_0 = 0$. We also have

$$\begin{aligned} \frac{1+L(1-\theta)\Delta t}{1-L\theta\Delta t} &= 1 + \frac{L\Delta t}{1-L\theta\Delta t} \\ &\leq \exp\left(\frac{L\Delta t}{1-L\theta\Delta t}\right) . \end{aligned}$$

In turn this means

$$\begin{aligned} \left(\frac{1+L(1-\theta)\Delta t}{1-L\theta\Delta t}\right)^n &\leq \left(\exp\left(\frac{L\Delta t}{1-L\theta\Delta t}\right)\right)^n \\ &\leq \exp\left(\frac{nL\Delta t}{1-L\theta\Delta t}\right) \\ &\leq \exp\left(\frac{LT}{1-L\theta\Delta t}\right) . \end{aligned}$$

Thus we have

$$|e_n| \leq \frac{T_{\max}}{L}\left[\exp\left(\frac{LT}{1-L\theta\Delta t}\right) - 1\right] , \tag{10}$$

for $n = 0, 1, \ldots, N$. This shows that the pointwise error has the same order as the truncation error.

3 Implementation

Recall that the θ-method is

$$\frac{U_{n+1} - U_n}{dt} = \theta f(t_{n+1}, U_n + 1) + (1-\theta) f(t_n, U_n)$$

If $\theta \neq 0$ then we have an implcit equation to solve for $U_n + 1$ at each timestep. We can write this equation as

$$g(U_{n+1}) := U_{n+1} - U_n - dt\theta f(t_{n+1}, U_n + 1) - dt(1-\theta) f(t_n, U_n) = 0 .$$

We can solve this using the Newton Rhapson method which is summarized in the code below.

```
% file mynewt.m
% this function finds a root of f(x) using Newton's method and a starting
function x=mynewt(f,fprime,xguess,tol)

x=xguess;

while abs(f(x)) > tol
    x=x-f(x)/fprime(x)
end
end
```

4 Numerical Example

Consider the specific problem

$$\frac{\mathrm{d}u}{\mathrm{d}t} = loglog(4+u^2)$$

for $0 < t \leq 1$ and with $u(0) = 1$. The numerical results are shown in Figure (1) below.

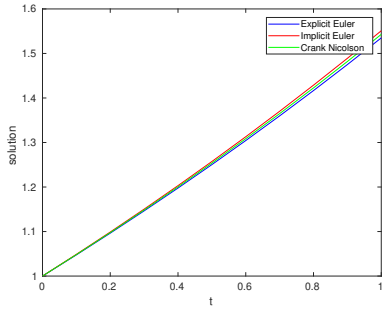

Figure 1: Numerical solution to the example problem.

4.1 Convergence results

Since the exact solution to this problem is not known, we use a very accurate solution generated using the Crank Nicolson scheme with $N = 10000$ to simulate the exact solution. We then consider the error at time $t = 1$. The results are shown in the figure. We can see that the errors for implicit and explicit Euler are almost the same and converge like $\mathcal{O}(\Delta t)$, whereas the implicit Euler scheme is $\mathcal{O}(\Delta t^2)$.

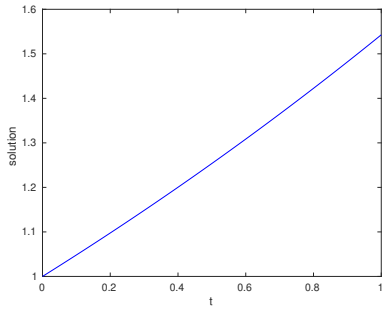

Figure 2: "Exact" solution to the example problem.

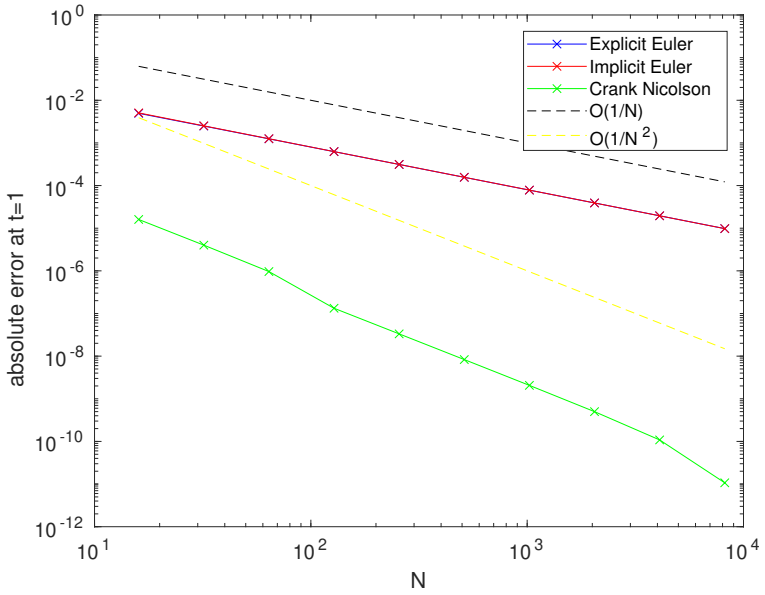

Figure 3: Convergence to the exact solution of the example problem at time $t = 1$. The errors for implicit and explicit Euler are almost the same and have size $\mathcal{O}(\Delta t)$, whereas the Crank Nicolson error is $\mathcal{O}(\Delta t^2)$.

5 Conclusion

We have looked at the θ-method for solving initial value ordinary differential equation problems. The parameter θ is chosen to lie in the interval $[0, 1]$. If $\theta = 0$ then the numerical method is explicit, otherwise it is implicit and a nonlinear equation must be solved at each timestep. If $\theta = \frac{1}{2}$ the method is second order accurate, otherwise the method is first oder accurate. These convergence rates were demonstrated numerically.

References

[1] K.W. Morton and D.F. Mayers. Numerical Solution of Partial Differential Equations. Cambridge University Press, 1994.

[2] Süli, E. & Mayers, D. F. An Introduction to numerical analysis.