Numerical Solution of ODEs

Exercise Class

24th October 2023

Implicit One-Step Methods

Implicit Euler Implemented by `ieuler.m`:

$$\begin{aligned} \kappa_1 &= f(t, x + \tau\kappa_1), \\ \psi(t+\tau, t, x) &= x + \tau\kappa_1. \end{aligned}$$

Crank-Nicholson

$$\begin{aligned} \kappa_1 &= f(t, x), \\ \kappa_2 &= f\left(t+\tau, x + \frac{\tau}{2}\kappa_1 + \frac{\tau}{2}\kappa_2\right), \\ \psi(t+\tau, t, x) &= x + \frac{\tau}{2}\left(\kappa_1 + \kappa_2\right). \end{aligned}$$

Fixed Point

Computing κ_1 for the *Implicit Euler* method requires solving a potentially nonlinear equation. One method is via the use of a fixed point iteration: Compute the sequence $\{\kappa_1^{(n)}\}_{n\geq 0}$ with the iteration

$$\begin{aligned} \kappa_1^{(n+1)} &= f(t+\tau, x + \tau\kappa_1^{(n)}), \qquad n \geq 1, \\ \kappa_1^{(0)} &= f(t, x). \end{aligned}$$

Continue the iteration until

$$\left\|\kappa_1^{(n+1)} - \kappa_1^{(n)}\right\| \leq \texttt{TOL},$$

where `TOL` is a desired tolerance.

Newton's Method

As an alternative, we can also use Newton's method for solving the implicit equation (see `ieuler_newton.m`). Defining

$$\boldsymbol{F}(\kappa_1) = \kappa_1 - f(t+\tau, x + \tau\kappa_1),$$

we try to find a root of $\boldsymbol{F}(\kappa_1) = 0$, by defining the sequence $\{\kappa_1^{(n)}\}_{n\geq 0}$ as

$$\kappa_1^{(n+1)} = \kappa_1^{(n)} - \left(\frac{\partial \boldsymbol{F}}{\partial \kappa_1}(\kappa_1^{(n)})\right)^{-1} \boldsymbol{F}(\kappa_1^{(n)})$$

where, for $\kappa_1 \in \mathbb{R}^n$ and $\boldsymbol{F}(\kappa_1) = (F_1(\kappa_1), \ldots, F_n(\kappa_1))$, we define the *Jacobian* as

$$\frac{\partial \boldsymbol{F}}{\partial \kappa_1}(\kappa_1) = \begin{pmatrix} \frac{\partial F_1}{\partial \kappa_{1,1}}(\kappa_1) & \cdots & \frac{\partial F_1}{\partial \kappa_{1,n}}(\kappa_1) \\ \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial \kappa_{1,1}}(\kappa_1) & \cdots & \frac{\partial F_n}{\partial \kappa_{1,n}}(\kappa_1) \end{pmatrix}.$$

Note, that

$$\frac{\partial \boldsymbol{F}}{\partial \kappa_1}(\kappa_1) = I - \tau f_x(t+\tau, x+\tau\kappa_1),$$

where f_x is the first derivative of f with respect to the second argument.

Convergence Analysis

Theorem 1. *Let there exist a positive constant C such that the local discretisation error is bounded by*

$$d(t+\tau, t, u(t)) \le C\tau^{p+1},$$

for all $\tau \le \tau_1$, $t \in [t_0, T]$. Consider an equidistant partition $\{t_j\}_{j=0}^N$ and approximate solution $\{u_j\}_{j=0}^N$, where

$$u_0 = x_0, \qquad u_{j+1} = \psi(t_{j+1}, t_j, u_j), \qquad j = 0, \dots, N-1.$$

Then,

$$\|u(t_j) - u_j\| \le \frac{e^{\Lambda(t_j - t_0)} - 1}{\Lambda} C\tau^p, \qquad j = 1, \dots, N.$$

Here, p is the order of the method.

If we study the error at the last time step

$$\underbrace{\|u(T) - u_N\|}_{\mathcal{E}_N} \le \underbrace{\frac{e^{\Lambda(T-t_0)} - 1}{\Lambda} C}_{K \text{ — constant}} \tau^p;$$

then,

$$\log_{10} \mathcal{E}_N \le \log_{10} K + p \log_{10} \tau.$$

Hence, we should observe asymptotically as $\tau \to 0$ that

$$\log_{10} \mathcal{E}_N = q + p \log_{10} \tau,$$

where $q = \log_{10} K$ is a constant.

Exercises

1. Modify `compare.m` to compare Euler (`eul.m`), Implicit Euler using a fixed point iteration (`ieuler.m`), and Implicit Euler using a Newton iteration (`ieuler_newton.m`), for solving the ODE

$$x'(t) = \begin{pmatrix} 998 & 1998 \\ -999 & -1999 \end{pmatrix} x(t), \qquad t \in [0, 0.1], \tag{1}$$

$$x(0) = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \tag{2}$$

   (`linsystem.m` and `linsystem_newton.m`) with $\tau = 0.002, 0.0021, 0.0019$. Also try with smaller values of τ, such as $\tau = 0.0001$ to try to reduce oscillations in the numerical solution.

   Remark. Make sure to print and check the values obtained from the solver, some of these methods will return `NaN` (Not a Number) values.

2. Implement the Crank-Nicholson method as a MATLAB function using a fixed point iteration, and test for the linear system (1)–(2).

3. Modify `one_step_order.m` to calculate the order of the Runge, Runge-Kutta, Heun, implicit Euler, and Crank-Nicholson methods, using the logistic equation

$$\begin{aligned} x'(t) &= (a - bx(t))x(t), \qquad && t \in [0,2], \\ x(0) &= x_0, \end{aligned}$$

with $a = b = 1$, $x_0 = 2$, and known exact solution

$$x(t) = \frac{x_0 e^t}{1 - x_0(1 - e^t)}.$$

Remark. Note that the implicit Euler method `ieuler` may not converge for $\tau = 1/2$. Therefore, the convergence analysis code needs to be changed to start from $\tau = 1/4$.