Part I. Theory

Problem 1. Consider the theta method,

\[ y_{n+1} = y_n + h \left[ \theta f(t_n, y_n) + (1 - \theta) f(t_{n+1}, y_{n+1}) \right]. \]

a. Find the domain of linear stability \( D_\theta \subset \mathbb{C} \). Sketch this in the complex plane for \( \theta = 1/3 \) and \( \theta = 2/3 \).

b. For which values \( \theta \in [0, 1] \) is the method \( A \)-stable? Give a proof.

Problem 2 (Iserles, Exercise 2.4). Determine the order of the three-step method

\[ y_{n+3} - y_n = h \left[ \frac{3}{8} f(t_{n+3}, y_{n+3}) + \frac{9}{8} f(t_{n+2}, y_{n+2}) + f(t_{n+1}, y_{n+1}) + \frac{3}{8} f(t_n, y_n) \right], \]

the \textit{three-eighths} scheme. Is it convergent?

Part II. Programming

Download the sample code \texttt{hw2.py}, which contains implementations of the Euler and backward Euler methods for \textit{systems} of ODEs. (Last week’s code was only designed for scalar ODEs.) The backward Euler code uses the nonlinear root-finder \texttt{fsolve} from the \texttt{scipy.optimize} library to solve for \( y_{n+1} \) at each step. Specifically,

\[ y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}) \iff 0 = -y_{n+1} + y_n + hf(t_{n+1}, y_{n+1}), \]

so at each step, we can use \texttt{fsolve} to find a root of the function

\[ F(y_{n+1}) = -y_{n+1} + y_n + hf(t_{n+1}, y_{n+1}), \]

with initial guess \( y_n \).

Problem 3. Consider the simple harmonic oscillator

\[ x'' = -x, \]

which can be written as the first-order linear system

\[ \begin{pmatrix} x' \\ v' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} x \\ v \end{pmatrix}. \]

a. Apply the Euler method to this problem for \( t \in [0, 100] \), with initial condition \( \begin{pmatrix} x_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \). Plot \( x(t) \) vs. \( t \) for \( h = 0.1 \) and \( h = 0.01 \).

b. Repeat part (a) for the backward Euler method.

Problem 4. Create a function \texttt{trapezoid(f, t0, y0, h, N)} that implements the trapezoid method. (Hint: This function should be very similar to \texttt{backwardEuler}, but with a different choice of \( F \).) Use this to repeat Problem 3(a) for the trapezoid method.