Homework 2

See due date on Canvas

COE 347

There is absolutely no tolerance for academic misconduct. All assigned material is to be prepared individually. Late assignments will not be accepted, unless under exceptional circumstances at the instructor's full discretion.

Submit your homework electronically as a PDF via Canvas by 11:59pm on the due date.

If you are submitting a scanned copy of your handwritten notes, rather than a typeset document, please take the time to reduce the file to a manageable size by adjusting the resolution.

Objectives

The objective of this homework is to explore the application of numerical methods for the solution of ODEs with emphasis on accuracy of a numerical solution. You will consider all methods seen and discussed in class.

Instructions

1. **80 pts.** Consider the following ODE

$$\frac{dy(x)}{dx} = y'(x) = -50(y - \cos(x)). \tag{1}$$

Integrate the ODE with initial conditions $y(0) = 0$ over the interval $x \in [0, 1]$ using MATLAB built-in functions. The solution will serve as your "exact" solution.

- **25 pts.** Implement explicit Euler, implicit Euler, midpoint, trapezoidal, Adams-Bashforth 2, explicit Runge-Kutta 2 (RK2), and explicit Runge-Kutta 4 (RK4) into MATLAB solvers. Details for the explicit RK4 method are given below and are also available on the class notes (Chapter 4).

  Always start with a small value of h and plot the numerical solution in $[0, 1]$ together with the MATLAB solution above. Confirm that you implemented the method correctly and get an idea of suitable time step sizes.

  Turn in plots of numerical solutions on the same graph together with the exact solution. For each method, produce a separate plot with the numerical solutions obtained with four reasonable choices of the time step size h as to convey the idea that the numerical solution is *converging to the exact one*.
- **15 pts.** For each method, advance the solution until $x = 1$. Repeat with various step sizes h and compute the difference between the numerical solution and the "exact" solution. Call this difference the (global) "error" for a given step size, $e(h)$.

  Plot the absolute value of the error $|e|$ (y-axis) as a function of $1/h$ (x-axis), where h is the step size. Use a log-log plot and report the error for all numerical methods considered.

- **15 pts.** For each method, fit a function $E = Ch^{\alpha}$ to the $(h, |e|)$ pairs and confirm that α is close to the value you expect given the rate of convergence of a specific method from the book and class notes.
- **25 pts.** For each method and a choice of time step h, produce a measure of the "work" (i.e. the "effort") required to integrate the IVP from $t = 0$ to $t = 1$. One sensible manner of measuring "work" is to count *how many times the function* $f(t, y)$ *is evaluated* as the method steps from $t = 0$ to $t = 1$. Modify the MATLAB functions to accomplish this objective (e.g. you may do this by updating a global counter every time the function is called) and call this number of function evaluations M.

  Produce another log-log plot, whereby you plot e vs. M for all numerical methods and time step sizes considered. In other words, your plot will show e vs. M as h decreases (and M increases) for all methods.

  What can you conclude by comparing the curves obtained with the various methods?

  A point worth considering. When you implement an implicit method, evaluations of $f(t, y)$ *are required to solve for the implicit equation that has* y_{n+1} *as a solution. So, to an extent, the function responsible for solving the implicit equation (e.g. the MATLAB function* `fsolve`*) decides how many function evaluations are required at each time step in order to achieve a user-prescribed tolerance.*

2. **20 pts.** Consider the following $N \times N$ tridiagonal matrix T_N:

$$T_N = \begin{bmatrix} -2 & 1 & & 0 \\ 1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 1 & -2 \end{bmatrix} \tag{2}$$

where $N \geq 1$.

Compute the eigenvalues of the matrix for $N = 10$ and confirm that they are

$$\lambda_i = -2(1 - \cos(\pi i/(N + 1)) \qquad i = 1, \ldots, N \tag{3}$$

Next, plot $\max |\lambda|$, i.e. the maximum absolute value of all eigenvalues of T_N versus N for $N = 1, \ldots, 20$. Comment on the behavior of $\max |\lambda|$ as N changes. Reconciliate your finding with the expression for λ_i provided above in Eq. (3).

Turn in plots that support your answers and explanations.

Runge Kutta, 4th order (RK4)

$$\begin{aligned} k_1 &= f(x_n, y_n) & (4) \\ k_2 &= f(x_n + h/2, y_n + (h/2)k_1) & (5) \\ k_3 &= f(x_n + h/2, y_n + (h/2)k_2) & (6) \\ k_4 &= f(x_n + h, y_n + hk_3) & (7) \\ y_{n+1} &= y_n + h[(1/6)k_1 + (2/6)k_2 + (2/6)k_3 + (1/6)k_4]. & (8) \end{aligned}$$